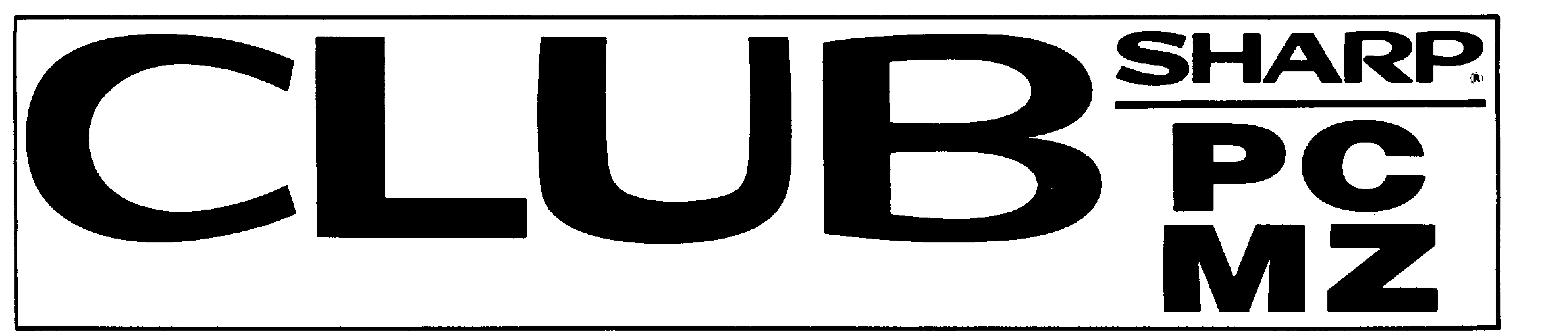


AVERTISSEMENT

J'ai rencontré l'autre jour un vieux journaliste qui m'a dit que les éditoriaux n'étaient en général pas ou très peu lu.

Or en lisant le premier jet de ce bulletin nouvelle formule (chose promise... chose due !), je constate que nos amis des clubs ne connaissent pas de vieux journalistes puisque j'ai découvert :
- un éditorial PC 1211
- un éditorial PC 1500
- un éditorial MZ 80 (et encore il paraît que Sylvain BIZOIRRE en avait préparé un pour le K, un pour le A et un pour le B).

De plus je crois savoir que Luc BURELLER est déjà en train de préparer celui du PC 1251 pour le prochain bulletin.

Pour réagir à cet état de chose, j'ai décidé de supprimer dans ce bulletin mon éditorial. A la place vous ne trouverez donc ce trimestre que le MOT de SHARP.

D. MONTERNOT.

Le mot de SHARP

Voici un bulletin en habit de gala qui je l'espère vous procurera beaucoup de plaisir.

Vous y trouverez les rubriques habituelles, mais désormais largement développées grâce à l'arrivée de nouveaux « journalistes ».

Je profite donc de ces colonnes pour citer et remercier tous ceux qui ont collaboré à cette édition.

Christophe CAMUS, Frédéric SANGUINETTI, Olivier BOISSE, (PC 1500/PC 1211) et Simon CHAGNOUX (MZ).

Sans oublier bien sûr les deux piliers du Club Luc BURELLER et Sylvain BIZOIRRE.

Au hasard d'un coup de téléphone ou d'un courrier c'est certainement l'un d'entre eux qui vous répondra et vous aidera.

Puisque nous en sommes au chapitre des félicitations j'aimerais encore citer quelques personnes.

Tout d'abord TROIS SHARPENTIERS qui nous ont envoyés des articles très intéressants Bernard MILLET (Soyez logique);, Christophe BOUCHON (Clavier) et J.C. LAURENT (Astuces) et puis bien sûr tout ceux qui envoient des programmes qui viennent enrichir les logithèques.

Un petit problème pourtant il semble que les PECEQUINZECENTISTES ne soient pas très courageux...

WAEL KOMBAR lui a gagné avec sa PC 1211 le concours OTHELLO organisé par notre confrère (pourquoi pas !) l'ORDINATEUR INDIVIDUEL et ceci face à 70 autres PICO-ORDINATEURS, notre fierté n'est pas des moindres lorsque l'on sait que dans ce concours il n'y avait que 3 PC 1211 engagés et que deux d'entre elles se classent dans le trio de tête.

Enfin pour clore les éloges et parce qu'on n'est jamais si bien servi que par soi-même un grand merci à SBM pour les moyens qu'elle met à la disposition des clubs.

Voilà je pense n'avoir oublié personne dans ce tableau d'honneur du club, et je me fais l'interprète de SHARP auprès de chacun d'entre vous pour vous souhaiter d'excellentes fêtes de fin d'année.

Et une année 1983 SHARPENTIEREMENT Réussie.

D.M.

Réassignation du clavier et des caractères du PC-1500

Le PC-1500 à deux possibilités non déclarées dans le manuel : on peut réassigner toutes les touches du clavier (sauf DEF, SML, SHIFT et ON/BREAK), c'est-à-dire changer le caractère obtenu en appuyant sur une touche ou sur touche SHIFTÉE (les fonctions MODE, RCL, FI...F6, ENTER, OFF, CL, CA, INS, DEL, SHIFTMODE et sont considérées comme des caractères dont les codes ASCII sont compris entre & 00 et & 1 F.

Le PC-1500 lorsqu'il reçoit les coordonnées d'une touche qui a été enfoncée au clavier convertit ces coordonnées en un code ASCII par l'intermédiaire d'un tableau : chaque couple de coordonnées possibles correspond à deux éléments dans le tableau un pour la touche pressée seule, un pour la touche précédée de SHIFT, chaque élément de ce tableau est un octet en ROM qui donne le code ASCII voulu. Il y a 64 touches au clavier (plus la touche ON/BREAK mais celle-ci ne fonctionne pas de la même façon) donc le tableau fait 2 x 64 = 128 octets : dans la ROM du BASIC, il va de l'adresse & FE80 à & FEFF. Bien sûr, tant que ce tableau est en ROM, on ne peut pas le modifier. Il faut pour cela faire savoir au PC-1500 qu'on désire que le tableau soit lu dans la RAM utilisateur (&4000 à &47FF pour le PC-1500 seul, &4000 à &47FF avec le CE-151 et &3800 et &5FFF avec le CE-155). Soit &NN00 l'adresse qui suit immédiatement le 128ème octet où l'on veut placer le nouveau tableau, on fait alors POKE & 785D, & 80, & NN pour indiquer l'adresse du nouveau tableau au PC-1500. Mais l'ancien clavier est encore actif. Pour activer le nouveau, on fait : POKE&764E, &4OR PEEK&764E.

Pour revenir à l'ancien, on fait simplement : SML (Appuyer sur la touche SML). Si vous n'avez pas le CE 155, vous ferez : NEW &4200 (Pour protéger de la mémoire contre l'effacement par la commande NEW seule), POKE &785D, &80, &42.

Editorial 1500

Avant toute chose nous vous demandons de bien vouloir nous excuser pour le retard que nous avons pris dans le courrier. Tout a été enregistré ne vous inquiétez plus les réponses sont en cours.

C'est à partir de ce bulletin que le club PC 1500 commence réellement à apporter de nombreux renseignements. Ceci c'est grâce à vous qui nous avez donné des trucs et des astuces. Mais vous êtes encore trop peu nombreux à nous envoyer des programmes. N'oubliez pas que c'est votre bulletin, que c'est votre club.

Vous vouliez des codes, et bien nous vous donnons une partie de ceux que nous connaissons, à vous d'en tirer partie et de nous envoyer vos programmes en L.M.

Vous trouverez dans ce bulletin un exemple de présentation pour les programmes que vous nous ferez parvenir, en effet pour le 1500 les programmes recevront un certain nombre de bonus. Vous pourrez par conséquent recevoir autant de programmes que vous voudrez jusqu'à concurrence des bonus que vous aurez.

Un programme complet, c'est-à-dire :
- Listing
- K7 si status 2 ⩾2000 (nous vous la retournerons)
- explications : à quoi sert le programme, domaine d'application
- contenu des variables.
- explications : ligne à ligne des principales routines.

Vous donne d'ores et déjà droit à 2 bonus.

(Nous rappelons que le bonus qui est l'unité monétaire de logiciel club n'est pas côté en bourse).

Surtout expliquez bien les routines en L.M.

Certains d'entre vous ont cherché à développer des choses intéressantes autour de leur 1500 nous en parlerons dans une nouvelle rubrique courrier Annonce. Vous êtes très peu à nous avoir donné votre avis sur les clubs régionaux qu'en pensez-vous ? Ecrivez-nous.

Alors à vos plumes et à vos 1500.

A bientôt.

Luc BURELLER.

(Suite Réassignation)

Avec le CE-155 : NEW &3A00, POKE &785D, &80, &3A.

Dans le premier cas, les 128 adresses occupées par le tableau du nouveau clavier seront &4180 à &41FF, dans le second cas, ce seront les adresses mémoire &3980 à &39FF.

Cependant, il faut placer des données correctes dans ces 128 octets pour que le nouveau clavier ait une signification. Voici donc un tableau pour savoir à quelle touche correspond chacun des 128 octets.

ATTENTION : On ne peut pas modifier le caractère émis par les touches correspondant à une case contenant une étoile.

En fait, ce tableau est celui qui est contenu dans la ROM.

Pour entrer les valeurs que l'on a choisies, on utilise le petit programme suivant : 1 : FOR A = &3980 TO &39FF : INPUT « valeur »= , B : POKE A, B : NEXTA : END.

On peut aussi mettre les valeurs une à une par POKE (c'est plus long). L'adresse correspondant à une case du tableau se calcule ainsi : on cherche la touche dans le tableau, on prend sa « référence » (par exemple : Nesten &01, Pen &2D). L'adresse concernant la touche dans le tableau sera alors 3980 + référence.

Pour savoir ce que l'on peut mettre comme valeur dans chaque case et quel sera l'effet obtenu en appuyant sur la touche, consulter le tableau ASCII complet qui suit. Si on a pas le CE-155, remplacer les &39... par des &41 dans ce qui précède.

high \ low	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0	↑ ØB	N 4E	Y 59	SHIFT 01*	H 48	8 38	5 35	2 32	◆ Ø9	X 58	W 57	F1 11	S 53	OFF ØF	– 2D	. 2E
1	Ø 30	M 4D	U 55	F5 15	J 4A	7 37	4 34	1 31	ENTER ØD	(28	I 49	F6 16	K 4B	O 4F	L 4C	) 29
2	RCL 19	C 43	E 45	F2 12	D 44	/ 2F	* 2A	+ 2B	SPACE 2Ø	V 56	R 52	F3 13	F 46	P 5Ø	◀ Ø8	= 3D
3	SML 02*	Z 5A	Q 51	DEF 1B*	A 41	CL 18	MODE 1F	▶ ØC	↓ ØA	B 42	T 54	F4 14	G 47	9 39	6 36	3 33
4	√ 5B	n 6E	y 79	SHIFT Ø1*	h 68	8 38	5 35	2 32	◆ Ø9	x 78	w 77	! 21	s 73	OFF ØF	, 2C	. 2E
5	Ø 3Ø	m • 6D	u 75	% 25	j 6A	7 37	4 34	1 31	ENTER ØD	< 3C	i 69	& 26	k 6B	o 6F	l 6C	> 3E
6	RCL 19	c 63	e 65	" 22	d 64	? 3F	: 3A	◆; 3B	5E	v 76	r 72	# 23	f 66	p 7Ø	DEL 1D	@ 4Ø
7	SML 02*	z 7A	q 71	DEF 1B*	a 61	CA 1A	SHIFT MODE 1D	INS 1C	Π 5D	b 62	t 74	$ 24	g 67	9 39	6 36	3 33

high \ low	Ø	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0		SHIFT •	SML •						◀	◆	↓	↑	▶	ENTER		OFF
1		F1	F2	F3	F4	F5	F6	F7	CL	RCL	CA	DEF	INS	DEL	SHIFT MODE	MODE
2	SPACE	!	"	#	$	%	&		(	)	*	+	,	–	.	/
3	Ø	1	2	3	4	5	6	7	8	9	:	;	<	=	>	?
4	@	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O
5	P	Q	R	S	T	U	V	W	X	Y	Z	√	π	π		
6		a	n	c	d	e	f	g	h	i	j	k	ℓ	m	n	o
7	p	q	r	s	t	u	v	w	x	y	z					■
8	DEF SPACE	DEF A	DEF B	DEF C	DEF D	DEF E	DEF F	DEF G	DEF H	DEF I	DEF J	DEF K	DEF L	DEF M	DEF N	DEF O
9	DEF P	DEF W	DEF R	DEF S	DEF T	DEF U	DEF V	DEF W	DEF X	DEF Y	DEF Z	DEF SPACE	DEF SPACE	DEF =	DEF SPACE	DEF SPACE
A	C.P	C.P	C.P	C.P	C.P	C.P	C.P	C.P	C.P	C.P	C.P	C.P	C.P	C.P	C.P	C.P
B	C.P	C.P	C.P	C.P	C.P	C.P	C.P	C.P	C.P	C.P	C.P	C.P	C.P	C.P	C.P	C.P
C	C.P	C.P	C.P	C.P	C.P	C.P	C.P	C.P	C.P	C.P	C.P	C.P	C.P	C.P	C.P	C.P
D	C.P	C.P	C.P	C.P	C.P	C.P	C.P	C.P	C.P	C.P	C.P	C.P	C.P	C.P	C.P	C.P
E	PRE	PRE	PRE	PRE	PRE	PRE	PRE	PRE	PRE	PRE	PRE	PRE	PRE	PRE	PRE	PRE
F	PRE	PRE	PRE	PRE	PRE	PRE	PRE	PRE	PRE	PRE	PRE	PRE	PRE	PRE	PRE	PRE

* ce code ne peut être utilisé. PRE : octet de préfix des mots clés du BASIC inutilisables. DEF "touche" : agit comme la pression successive sur DEF puis sur la touche indiquée. C.P. : caractère créé par l'utilisation voir la suite de l'article.

Création de nouveaux caractères

Pour créer de nouveaux caractères, il suffit de mettre une table à partir de la fin de la table en RAM de réassignation clavier. Chaque nouveau caractère occupe 5 octets (codés comme pour G PRINT : 1 octet valant de ù à &7F pour chacune des 5 colonnes du caractère). Les codes ASCII des nouveaux caractères vont de &80F à &FF (on les obtient par CHR$) dont seuls ceux dont les codes vont de &A0 à &DF sont réassignables au clavier. Soit n le nombre de nouveaux caractères que l'on veut créer : on fait NEW &3A00 + nx5 avec le CE155 et NEW &4200 + nx5 sans CE-155, puis on exécute le programme suivant :

1 : INPUT « N », N : FOR A= OTO N * 5—1 : INPUT B : POKE &A00 + A, B : NEXT A : END.

Donner le nombre de caractère voulu puis les données les constituants un par un. Remplacer &3A00 par &4200 si vous n'avez pas le CE-155.

Quelques caractères sont donnés comme exemples.

Pour mettre sur cassette les deux tableaux, on fait :

avec CE-155 : CSAVEM « CLAVIER » ; &3980,&39 FF

CSAVEM « CARACTERES », &3A00, &3A00 + n x 15 - 1.

sans le CE-155 : idem mais &39... devient &41... et &3A... devient &42...

Pour les relier : CLOADM « CLAVIER » ou CLOADM « CARACTERS ».

↑	=>	4	2	127	2	4
→	=>	8	8	42	28	8
↓	=>	16	32	127	32	16
←	=>	8	28	42	8	8
↗	=>	64	32	20	12	28
↘	=>	4	8	80	96	112
↙	=>	112	96	80	8	4
↖	=>	28	12	20	32	64
[illegible]	=>	114	49	85	70	39
A	=>	30	5	5	30	0
B	=>	31	21	21	10	0
C	=>	14	17	17	10	0
D	=>	31	17	17	14	0
E	=>	31	21	21	17	0
F	=>	31	5	5	1	0
G	=>	14	17	21	13	0
H	=>	31	4	4	31	0
I	=>	0	17	31	17	0
J	=>	8	17	15	1	0
K	=>	31	4	10	17	0
L	=>	0	31	16	16	0
M	=>	31	2	4	2	31
N	=>	31	2	4	31	0
O	=>	14	17	19	15	0
P	=>	31	5	5	2	0
Q	=>	14	17	25	30	0
R	=>	31	5	13	18	0
S	=>	18	21	21	9	0

T	=>	0	1	31	1	0
U	=>	15	16	16	15	0
V	=>	3	12	16	12	3
W	=>	31	8	4	8	31
X	=>	0	27	4	27	0
Y	=>	17	10	4	3	0
Z	=>	18	26	22	18	0
0	=>	14	17	17	14	0
1	=>	0	18	31	16	0
2	=>	18	25	21	18	0
3	=>	17	21	21	10	0
4	=>	12	10	31	8	0
5	=>	23	21	21	9	0
6	=>	12	22	21	8	0
7	=>	1	29	3	1	0
8	=>	10	21	21	10	0
9	=>	2	21	13	6	0
+	=>	0	4	14	4	0
-	=>	0	4	4	4	0
×	=>	0	10	4	10	0
/	=>	0	8	4	2	0
^	=>	0	4	2	4	0
(	=>	0	14	17	0	0
)	=>	0	0	17	14	0
±	=>	0	18	23	18	0
-1	=>	4	4	18	31	16
°	=>	7	5	7	0	0
'	=>	0	5	3	0	0
"	=>	5	3	0	5	3
A	=>	120	20	20	120	0
B	=>	124	84	84	40	0
C	=>	56	68	68	40	0
D	=>	124	68	68	56	0
E	=>	124	84	84	68	0
F	=>	124	20	20	4	0
G	=>	56	68	84	52	0
H	=>	124	16	16	124	0
I	=>	0	68	124	68	0
J	=>	32	68	60	4	0
K	=>	124	16	40	68	0
L	=>	0	124	64	64	0
M	=>	124	8	16	8	124
N	=>	124	8	16	124	0
O	=>	56	68	76	60	0
P	=>	124	20	20	8	0
Q	=>	56	68	100	120	0
R	=>	124	20	52	72	0
S	=>	72	84	84	36	0
T	=>	0	4	124	4	0
U	=>	60	64	64	60	0
V	=>	12	48	64	48	12
W	=>	124	32	16	32	124
X	=>	0	108	16	108	0
Y	=>	68	40	16	12	0
Z	=>	72	104	88	72	0
0	=>	56	68	68	56	0
1	=>	0	72	124	64	0
2	=>	72	100	84	72	0
3	=>	68	84	84	40	0
4	=>	48	40	124	32	0
5	=>	92	84	84	36	0
6	=>	48	88	84	32	0
7	=>	4	116	12	4	0
8	=>	40	84	84	40	0
9	=>	8	84	52	24	0
10	=>	120	0	120	72	120
±	=>	68	68	95	68	68
≤	=>	80	88	84	82	80
≠	=>	20	52	28	22	20
≥	=>	80	82	84	88	80
※	=>	28	34	28	34	28
∫	=>	32	64	62	1	2
X̄	=>	69	37	57	73	69
α	=>	24	36	36	24	36
β	=>	64	63	21	22	8
Γ	=>	127	1	1	1	3
δ	=>	49	74	68	74	49
Δ	=>	24	20	18	20	24
ε	=>	48	76	74	74	32
θ	=>	60	74	74	61	0
λ	=>	66	36	24	32	64
μ	=>	64	60	16	28	32
ρ	=>	64	62	9	9	6
Σ	=>	99	85	73	65	99
σ	=>	24	36	36	28	4
Φ	=>	8	85	127	85	8
φ	=>	6	8	126	9	6
Ψ	=>	7	8	127	8	7
Ω	=>	88	100	4	100	88
Å	=>	56	69	70	69	64
£	=>	64	56	68	68	40
é	=>	56	84	86	85	24
è	=>	56	85	86	84	24
ê	=>	56	86	85	86	24
ï	=>	0	1	124	1	0
Æ	=>	56	68	60	84	88
ô	=>	56	70	69	70	56
ö	=>	56	69	68	69	56
ü	=>	60	65	66	32	124
►	=>	127	62	28	8	0

Astuce

Effet de status — Effet de RND

Dans la catégorie « instruction mal expliquée » nous allons voir aujourd'hui les instructions STATUS et RND

Commençons par l'ordre STATUS.

Dans le manuel deux cas sont expliqués, en fait il y en a bien d'autres. Cette instruction s'utilise sous la forme status n, avec 0 ≤ n ≤ 255 et signifie selon la valeur de n :

si n = 0 :

La commande renvoie la taille mémoire utilisateur encore libre c'est-à-dire le nombre de pas de programme utilisables.

si n = 1 :

Ici le nombre donné représente la taille du programme qui se trouve en mémoire (ou de tous les programmes après un merge) STATUS 0 et STATUS 1 sont en fait des compléments l'un de l'autre par exemple dans un PC 1500 sans module mémoire STATUS 0 + STATUS 1 est toujours égal à 1850.

si n= 2 :

On obtient l'adresse mémoire + 1 de la zone mémoire programme. On remarquera (voir explication § ligne de prg) que PEEK (Status 2—1) donne toujours 255.

si n = 3 :

On obtient l'adresse mémoire —1 de la zone mémoire donnée. Ceci est très utile car si dans un programme où il y a des dimensionnements de variables on veut connaître le nombre de pas programmables effectivement il faut faire (STATUS 3 — STATUS 2) et non pas STATUS 0.

si n ≥ 4 :

Le 1500 nous donne le numéro de la dernière ligne exécutée. Après un end ou si aucun programme n'a été exécuté STATUS 4 vaut 0.

Toutes les formes de STATUS n sont programmables et sont très utiles dans l'utilisation des variables ou bien dans les goto relatifs (sous programmes).

La fonction RND (n) nous renvoie un nombre aléatoire mais attention car ce nombre aléatoire dépend en quelque sorte de n.

En réalité 3 cas se présentent :

si n > 0 :

Alors RND (n) donne un nombre aléatoire entier compris entre 1 et n.

1 ≤ RND (n) ≤ n

si n = 0 :

Alors RND (0) donne un nombre aléatoire décimal compris entre 0 et 1

0 < RND (0) < 1

si n < 0 :

Alors on définit une fonction pseudo-aléatoire d'argument n.

RANDOM RND (—5)

Désormais en RND (n) avec n ≥ 0 on aura une suite de nombre aléatoire que l'on retrouvera toujours après ou RND (—5).

Une ligne de programme

Lorsque l'on introduit une ligne de programme celle-ci vient s'inscrire quelque part dans la mémoire et ce d'une certaine manière, c'est ce que l'on va voir maintenant.

L'instruction STATUS 2 lorsque la mémoire est vide nous donne l'adresse + 1 de la zone mémoire programme, c'est à partir de cette adresse que vont venir s'inscrire les lignes de programme que l'on introduira.

Du côté utilisateur, une ligne de programme, c'est un numéro, une ou plusieurs instructions avec leurs données et la pression de la touche ENTER pour valider cette ligne.

Du côté machine cela se passe à peu près de la même manière. Tout d'abord un numéro codé sur deux octets (rappelons que le mot octet correspond au mot pas et qu'1 Kilo octet (K0) équivaut à 1024 octets. De ce fait on en déduit facilement la valeur du plus grand numéro de ligne : un octet contient au maximum 255 si les 2 valent 255 on a en décimal le numéro 255x 256 + 255 c'est-à-dire 65535, or la valeur 255 a une signification spéciale et ne peut en aucun cas représenter le premier octet d'un numéro de ligne. Donc le numéro maximum sera :

254 x 256+ 255 c'est-à-dire 65279.

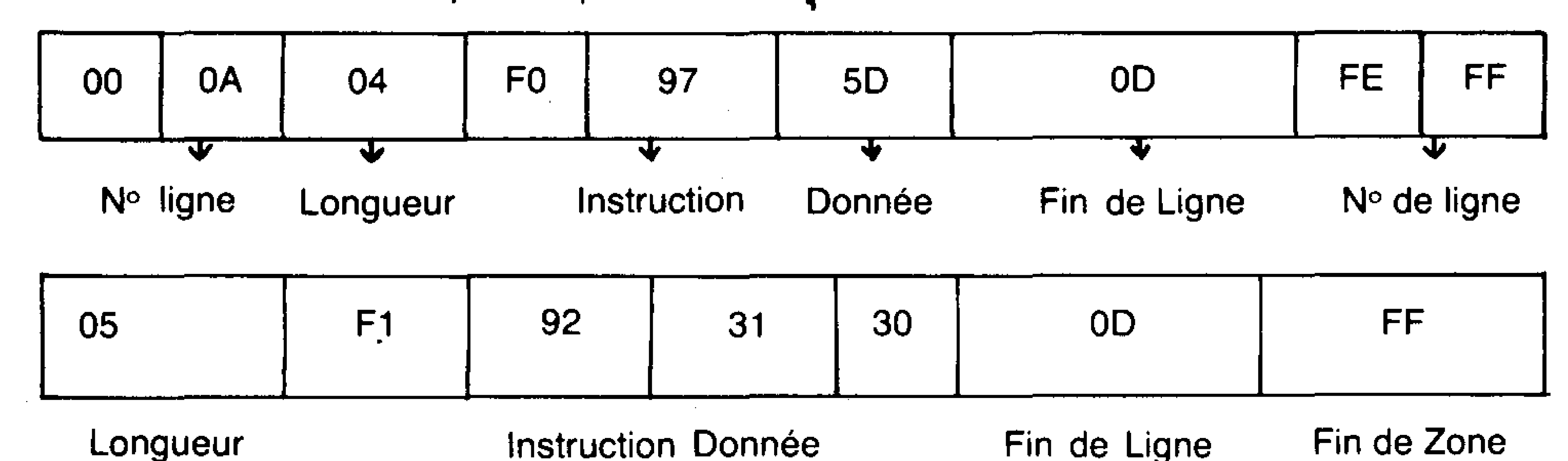
Ensuite il y a un octet donnant la longueur de la ligne et la ligne elle-même codée en ASCU sauf pour les mots réservés qui sont

codés sur 2 octets. Et puis le code $0D qui représente ENTER et qui est le code de fin de ligne. A cet instant 2 cas se présentent : s'il y a une autre ligne de programme derrière la codification reprend immédiatement.

S'il n'y a pas d'autre ligne derrière celle-ci la machine met un &FF dans l'octet qui suit le dernier OD. Cette valeur est une sorte de verrou qui indique que la zone de mémoire programme se termine momentanément ici. Etant donné que ce &FF ferme toujours la marche c'est la raison pour laquelle PEEK (STATUS 2—1) = &FF.

Voyons ceci avec 2 exemples : faites
- NEW0
- STATUS 2 → 16581 (si PC-1500 seul ou PC+ CE151) 14533 (si PC + CE 155).
- 1 0 PRINT π ENTER
- 65279 GOTO 10

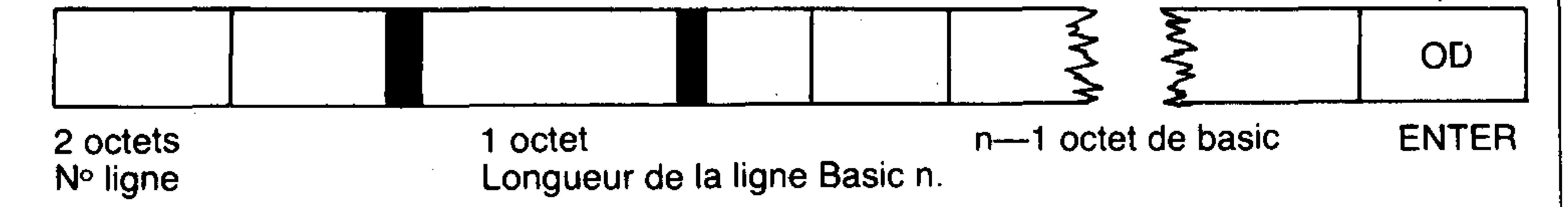
Le status 2 nous a permis de connaître l'emplacement de la zone mémoire désormais par PEEK successif à partir de cette adrsse vous pouvez retrouver la structure des lignes introduites.



10 : PRINT π ENTER 65279 : GOTO 1 0 ENTER FIN

soit sous une forme plus générale.

Nous aurons sur l'afficheur :



Voici un petit programme amusant utilisant la mémoire programme. Il permet de mettre dans un print le contenu d'une variable. Faire NEW0.

ATTENTION si PC + CE155 mettre I = &38CB en ligne 10.

```
10:PRINT "1234567
   89012345678901
   23456 ":I=&40C
   B:DIM A$(0)*26
20:INPUT "Texte:"
   ;A$(0):B=LEN (
   A$(0)):FOR J=I
   TO I+B:POKE J,
   ASC MID$ (A$(0
   ), J-I+1, 1):
   NEXT J
30:FOR J=I+BTO I+
   26:POKE J,&20:
   NEXT J:LLIST :
   END
```

Le listing doit être scrupuleusement respecté.

Des codes, la vidéo.

A l'occasion d'un programme présenté plus loin quelques codes LM seront présentés. En ce qui concerne le LM nous possédons environ 250 codes que nous avons découverts à force de recherche. Etant donné que nous nous sommes fabriqués un propre mnémonique pour nous faciliter l'utilisation de ce langage nous hésitons à donner un pseudo mnémonique empreint du Z80 c'est pourquoi les codes que nous vous donnerons seront expliqués mais ne seront pas accompagnés d'un mnémonique.

Avant d'expliquer ce programme qui gère la ram vidéo voyons comment cette zone de la mémoire est constituée.

L'afficheur de la PC 1500 est constitué d'une matrice 7 x 156 et de 13 indicateurs. Commençons par le plus simple : les indicateurs au nombre de 13 ; ils représentent une fonction vérifiée s'ils sont allumés. Ainsi on peut changer de mode en changeant l'indicateur run par l'indicateur pro par exemple.

Les indicateurs représentent les bits composant les octets &764 E et &764 F.

Rappelons qu'un octet : une case mémoire, est constituée de 8 bits, 8 positions successives qui peuvent être mises à zéro ou à 1. C'est en fait la représentation binaire d'un nombre.

Ainsi en &764 E nous avons :

DEF I II III SM2 SHIFT BUSY

bit 7 6 5 4 3 2 1 0

et en &764 F nous avons :
RUN PRO RES RAD G DE

bit 7 6 5 4 3 2 1 0

Ainsi, si par poke on veut allumer, RESERVE, DE et RAD.

C'est-à-dire le bit 2 de l'octet &764 E et les bits 0,2 et 4 de l'octet &764 F on fera :

POKE &764E, 2 x 2, 2 x 0 + 2 x 2 + 2 x 4 ENTER

La matrice 7 x 156

La mémoire vidéo se trouve en 2 parties de 7600 à 764 D et de 7700 à 774D, elle occupe 156 octets de la mémoire système. Malheureusement cela aurait été trop simple si chacun de ses octets représentaient une colonne de l'afficheur.

Une colonne de l'afficheur voit sa représentation dans la mémoire sur deux octets et dans ces mêmes deux octets se trouvent 2 colonnes partagées dans les parties basses et hautes de cet octet.

Ainsi dans les octets d'adresse pair on trouve la partie haute de la colonne n+ 18 et n tandis que dans l'octet impair suivant on trouve la partie basse de la colonne n + 78 et n.

Représentons ceci par un schéma :

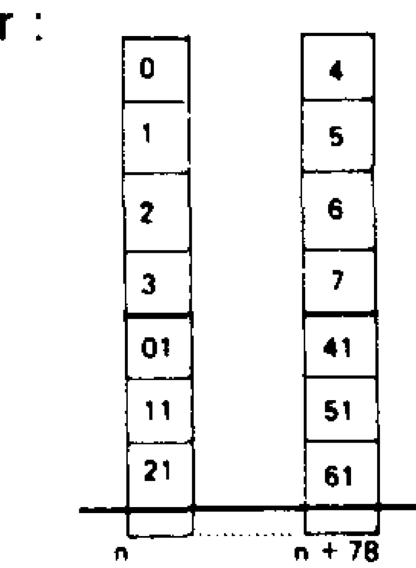
Octet d'adresse paire
7 6 5 4 3 2 1 0

Tous les bits servent.

Octet d'adresse impaire
7' 6' 5' 4' 3' 2' 1'

Les bits 7 et 3 ne servent pas.

Nous aurons sur l'afficheur :



Et de cette manière nous avons
Un bit à 1 correspond à un point allumé.
- de &7600 à &764 D
Les colonnes 78 à 116 et 0 à 38.
- de &7700 à &774 D.
Les colonnes 117 à 155 et 39 à 77.

Exemple : Si l'on veut allumer la partie haute de colonne 30 et la partie basse de la colonne 108 on fera :

POKE & 763c & 0F & 70.

Ecrire ceci dans un programme :
10 POKE & 763c, & 0F, & 70.
20 GOTO 20.

Maintenant vous pourrez vous attaquez au programme de voitures et essayé de comprendre à l'aide des explications la manière de rédiger un programme en L.M.

Passons à l'explication des instructions et famille d'instructions nécessaires à la compréhension de ce programme.

Le MPU possède en plus des registres exclusifs, sept registres utilisateurs.

Nous les nommerons A comme accumulateur et Uh, UL, Xh, XL, Yh, YL. qui sont les 6 autres registres utilisables par paire. (Lorsqu'ils seront utilisés comme double registre nous omettrons l'indice).

Passons maintenant à l'explication code à code.

Chargement d'un registre dans l'accumulateur : 1 octet.

0 4	A ← UL
8 4	A ← Uh
1 4	A ← YL
9 4	A ← Yh
2 4	A ← XL
A 4	A ← Xh

Chargement de l'accumulateur dans un registre : 1 octet.

0 8	Uh ← A
0 A	UL ← A
1 8	Yh ← A
1 A	YL ← A
2 8	Xh ← A
2 A	XL ← A

Chargement d'une donnée dans un registre : 2 octets.

4 8	Uh ← n
4 A	UL ← n
5 8	Yh ← n
5 A	YL ← n
6 8	Xh ← n
6 A	XL ← n
B 5	A ← n

Chargement du contenu d'un double registre dans l'accumulateur : 1 octet.

0 5	A ← (UU)
1 5	A ← (YY)
2 5	A ← (XX)

Chargement du contenu d'une adresse spécifiée par un double registre dans l'accumulateur et incrémentation ou décrémentation du 1 octet double registre utilisé.

1 octet.

4 5	A ← (UU)	UU ←UU + 1
4 7	A ← (UU)	UU ←UU — 1
5 5	A ← (YY)	YY ←YY + 1
5 7	A ← (YY)	YY ←YY — 1
6 8	A ← (XX)	XX ←XX + 1
6 7	A ← (XX)	XX ←XX — 1

Chargement de l'accumulateur à l'adresse spécifiée par un double registre suivi d'une opération sur ce dit registre.

1 octet.

41	(UU) ← A	UU ←UU + 1
43	(UU) ← A	UU ←UU — 1
51	(YY) ← A	YY ←YY + 1
53	(YY) ← A	YY ←YY — 1
61	(XX) ← A	XX ←XX + 1
63	(XX) ← A	XX ←XX — 1

Incrémentation décrémentation d'un double registre : 1 octet.

44	UU ←UU + 1
46	UU ←UU — 1
54	YY ←YY + 1
56	YY ←YY — 1
64	XX ←XX + 1
66	XX ←XX — 1

Test entre un registre et une donnée calcul r-n non conservé : 2 octets.

4C	Calcul Uh -n non conservé, comparé Uh et n.
4E	UL - n
5C	Yh - n
5E	YL - n
6C	Xh - n
6E	XL - n
B7	A - n

Saut relatif conditionnel ou inconditionnel 2 octets.

8 1 saut en avant de + e octet si résultat négatif
9 1 saut en arrière de —e octet si résultat négatif.
8 3 saut en avant de +e octet si résultat positif ou nul.
9 3 saut en arrière de 8e octet si résultat positif ou nul.
8 9 saut en avant de + e oxtet si résultat différent de zéro.
9 9 saut en arrière de —e octet si résultat différent de zéro.
8 B saut en avant de +e octet si résultat égal à zéro.
9 B saut en arrière de 8e octet si résultat égal à zéro.
8 E saut en avant de + octet.
9 E saut en arrière de —e octet.

Autre :
8 8 XL←XL — 1 et saut en arrière de —e octet si XL≥ 0.
2 octet
A E (nn)←A chargement de l'accumulateur à une adresse nn.
3 octets
B E CALL nn appel d'un sous programme d'adresse nn.
3 octets
9 A RTN retour d'un sous programme.
1 octet.
F 1 Echange la partie basse et la partie haute de l'accumulateur.
1 octet.

Et voilà, en espèrant que ces codes vous serviront, vous avez été nombreux à nous en demander. D'autres codes la prochaine fois.

Adresses intéressantes :

Un beep continu : Faites POKE &FDE7,n avec &41 ≤ n ≤&44.

Faites un beep de « fréquence » n pour l'arrêter ; éteindre la machine ou bien faire POKE &FDE7,0.

Tapez donc de petit programme amusant :

```
10:IF INKEY$ =""
   THEN 10
20:POKE# &FDE7,&4
   2
30:IF INKEY$ <>""
   THEN 30
40:POKE# &FDE7,0
50:GOTO 10
```

– Protégez un programme :
POKE # &F01D, &FF bloque le break s'il y a des wait 0 sur les print uniquement)
POKE # &F01D, &00 débloque le break.

– Obtenez de gros gros caractères sur l'imprimante. On n'est plus limité à 9 tailles de caractères mais à 36.

POKE &79F4, TAILLE
1≤ TAILLE≤36

– Eteignez votre machine par programme

CALL & CD 71

– Faites un clear en LM

CALL & D0 80

– Faites un beep CALL & E669 équivaut à beep 1.

– En LM pour faire des beeps mettez la fréquence dans XL, la durée sur deux octets dans UH et UL et faites & BE, & E6, &6F. Cela équivaut à BEEP 1, XL, Uh Ul.

– En LM scruter le clavier (équivaut à INKEY $)

& BE, & E4, & EC.

Au retour de ce sous programme on a l'ASCII de la touche enfoncée dans A si on a pressé une touche.

Logithèque SHARP PC-1500

	Pts BONUS
De M.GLUCK J.P. – « Calendrier perpétuel » sur K7 à renvoyer	3
De M. ARGO SAINT AVOLD Micro-informatique – Biorythme astral	4
De M. Cyril DEICHA – Répertoire téléphonique	5
De M. COLLANE – Le jeu de la puce	4
De M. Robert LOWY – Calendrier (opération sur les dates)	2
De M. Frédéric GUYON – Espérance de vie	3
De I.S – Pendu	3
– Complexes (calculs dans C)	8
De M. PERRE – Dump	4
De M. J.R. DELYS – Etudes graphiques	5

Programmes

1 — Programme de voitures :

Il s'agit d'un jeu où il vous faut parcourir 156 km sans accident pour que le jeu soit attrayant et assez rapide, la voiture : un point ne se déplace pas, mais c'est la route qui se déplace grâce à un programme en LM qui se charge de ceci.

Passons au programme tout d'abord : le basic.

Avant toute chose faites NEW & 38EF si vous avez un module 8K sinon faites NEW & 40EF et modifier les lignes.

90 : CALL & 40C6
180 : « D » RESTORE : FOR I=& 40C6 T0 & 40EE...
après def « D », faire POKE & 40C9, &40 ; POKE & 40D2, &40, et POKE & 40D7, &40.

L'entrée dans ce programme se fait de deux manières.
DEF « D » à faire la première fois obligatoirement.
DEF « » mise en fonction du prg jamais faire DEF « » ; après l'introduction du programme.

La partie basic du programme chargée de créer la route est assez lente mais attention le jeu commence dès que la route est finie.

Le but est d'atteindre la ligne d'arrivée qui vient vers vous par la droite. Vous possédez 10 vitesses : pavé numérique de 0 à 9 (plus la vitesse est petite plus vous allez vite), et 2 touches de direction ▲ et ▼ suivant que vous voulez monter ou descendre.

Attention si vous touchez les bottes de pailles c'est fini. Votre voiture se trouve toujours sous le D de deg.

Programme basic exploitation.

Ligne : 10 à 40 initialisations : création de la route.

050 Test si on passe en vitesse 0.
060 Test si le programme est fini avec victoire.
070 Changement de vitesse autre que 0 s'il y a lieu.
080 Extinction de la voiture.
090 Appelle au sous programme en L.M.
100 Test si on veut descendre.
110 Test si on veut monter.
120 Test si on s'est planté.
130 Affiche la voiture, boucle d'attente correspondant à la vitesse retour au programme.
140 à 150 Affichage de l'étoile signalant que l'on s'est tué au volant.
160 Félicitations si l'on a réussi le parcours.
170 Calcul des points-Affichage du Score.
180 Initialisation du programme L.M.
190 à 200 Data.

Contenu des variables :

A Bit bas de la route routine création.
B Valeur aléatoire routine création.
E Horaire au début de la partie et score.
F Flag signifiront si l'on a gagné ou perdu.
G Vitesse.
I Compteur de boucle.
M Flag test si l'on change de vitesse.
O,P,Q,R,S,T... Dessin de la route au niveau de l'arrivée.
U Horaire de fin de partie.
V Bit haut de la route... routine création.
W Position de la voiture (2 ^ Z) opératoire.
X Colonne affichée routine création.
z Position de la voiture (bit).

Programme en L.M. : Pour les explications détaillées des instructions voir (astuces).

Contenu des registres :

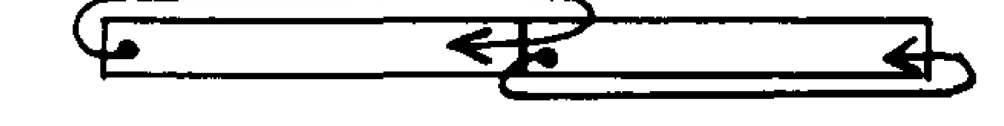
XL compte le nombre d'exécution, pour le décalage d'un pixel, il faut boucler deux fois sur ce programme d'où la valeur 01 au début du programme.

Uh-UL contient l'adresse sur laquelle on travaille ; son contenu est d'abord sauvé en 38C5 puis après le décalage resauve en YL.

Ye contient la valeur contenue à l'adresse UhUL et qui doit être décalée.

Explication ligne à ligne :

..C6 Compteur programmé utilise l'instruction 88 car le programme sera utilisé 2 fois (voir + loin.)
..C8 Appelle le sous programme qui se trouve en ..EA
..EA Met dans A ce qui se trouve dans la case vidéo la plus basse.
..ED Echange la partie haute et la partie basse de A. sinon le décalage écran se ferait en 2 moitiés.



..EE Retour au programme LM Principal.
..CB Sauve A dans Ye : mémoire contenu vidéo.
..CC Partie haute de la 2ème page vidéo.
..CE Partie basse de la 2ème page vidéo.
..DO On met dans A ce qui se trouve en Uh Ul (mémoire que l'on gère).
..D1 On le sauve en 38C4.
..D4 On rappelle dans A ce qui se trouvait en vidéo.
..D5 On la remet en vidéo (Pixel — 1/2) et on décrémente Uh Ul.
..D6 On remet dans A ce qui se trouve en 38C5.
..D9 On le met dans YL.
..DA On regarde si la page vidéo est finie.
..DC Si oui (UL = &FF) on saute 2 octets.
..DE Non alors on recommence.
..EO On regarde si les 2 pages ont été faites.
..E2 Si non (Uh = 75) on saute 3 octets.
..E4 Oui alors on regarde si le programme est terminé, on décrémente XL et si XL = &FF alors on recommence tout.
..E6 TERMINER, on retourne sous basic.
..E7 On décrémente Uh UL (changement de page).
..E8 On retourne dans le cœur du programme.

On boucle 2 fois sur ce programme à cause de la structure de la vidéo. Si on l'utilisait qu'une fois on retrouverait les parties basses et hautes de la vidéo inversée ex : A donne

Amusez-vous bien I.S.

```
10:" "V=5:Z=3:A=1
   :G=0:WAIT 0:
   FOR I=0TO 155:
   GCURSOR I:B=
   RND (3)-2:IF (
   A+B<0)OR (V+B>
   6)LET B=-B
20:IF I=54GPRINT
   &7F:NEXT I
30:A=A+B:V=V+B:
   GPRINT 2^A+2^V
   :IF I/3=INT (I
   /3)LET X=A+1+2
   *(RND (2)-1):
   GCURSOR I:
   GPRINT POINT I
   +2^X
40:NEXT I:BEEP 1:
   E=TIME :GOTO 9
   0
50:IF INKEY$ =
   CHR$ 48LET G=0
60:IF POINT 56=&7
   FLET F=0:U=
   TIME :GOTO 160
70:M=VAL INKEY$ :
   IF M<>0LET G=M
80:GCURSOR 55:
   GPRINT POINT 5
   5-W
90:CALL &38C6
100:IF INKEY$ =
   CHR$ 11LET Z=Z
   -1:GOTO 120
110:IF INKEY$ =
   CHR$ 10LET Z=Z
   +1
120:W=2^Z:IF (WAND
   POINT 55)=W
   THEN 140
130:GCURSOR 55:
   GPRINT POINT 5
   5+W:FOR I=0TO
   4*G:NEXT I:
   GOTO 50
140:U=TIME :F=1:
   FOR I=53TO 58:
   @(I-38)=POINT
   I:NEXT I
150:FOR I=1TO 10:
   WAIT 25:BEEP 1
   :GCURSOR 53:
   GPRINT O;P;Q;R
   ;S;T:GCURSOR 5
   3:PRINT "*":
   NEXT I
160:CLS :IF F=0
   WAIT 100:PRINT
   "Bravo !"
170:E=ABS (172+F*2
   06-(DEG U-DEG
   E)*10000):
   CURSOR 10:WAIT
   :PRINT "SCORE:
   ";USING "#####
   ";E:END
180:"D"RESTORE :
   FOR I=&38C6TO
   &38EE:READ A:
   POKE I,A:NEXT
   I:GOTO " "
190:DATA 106,1,190
   ,56,234,26,72,
   119,74,77,5,17
   4,56,197,20,67
   ,165,56,197,26
   ,78,255
200:DATA 139,2,158
   ,16,76,117,137
   ,3,136,30,154,
   70,158,28,165,
   118,0,241,154
```

Programmes « Musique » pour PC-1500 avec CE-151 et de préférence CE-150

Programme « Musique » pour PC 1500 avec CE 151 et de préférence CE-150

Ce programme permet d'introduire dans le PC-1500 toute sorte de partition, de les faire interpréter, et soit de les sauvegarder sur cassette, soit d'obtenir une liste de « DATA » permettant ensuite de sonoriser n'importe quel programme.

L'introduction de la partition se fera (c'est long !) note par note au clavier, (l'utilisation de la tablette CE-153 facilitera le travail) toutes les explications étant fournies par l'ordinateur aux lignes 50000-50030.

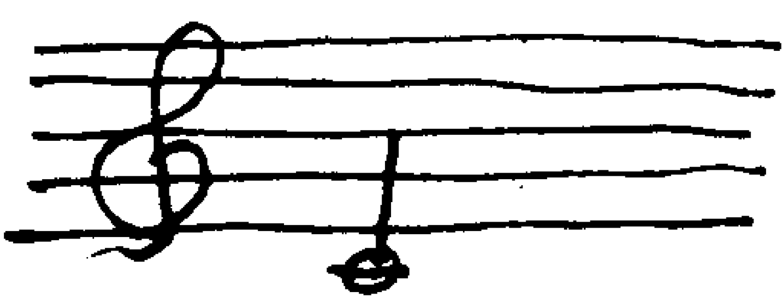
Lorsque toutes les notes sont entrées, SHARP nous donnera sa version de l'œuvre en question. Si quelques notes ne conviennent pas, il sera facile de les corriger après avoir répondu « N » à la question « satisfait ? ». Une fois les notes correctement introduites, vous pourrez modifier le temps, à chaque modification vous réentendrez la mélodie à la nouvelle vitesse. Quand tout sera parfait, vous pourrez alors enregistrer les données sur cassette (çà aussi, c'est long !) ou (et c'est je crois le plus utile) en avoir la liste sur papier.

Bonne musique
Sharpement vôtre
C. DUHAMEL

Explications :
F (I) contient les fréquences.
D (I) les durées.

Je me suis contenté de DIM (100), mais on peut faire mieux.

De 50 à 120, on introduit le nombre de 1/2 tons séparant la note à jouer de la note la plus grave (ici le do sous les lignes) :



– de 130 à 190, on introduit la durée
– en 200 et 210, on calcule F et D.

Vous avez certainement remarqué qu'une valeur constante de la durée ne donnait pas toujours la même longueur de beep (elle dépend également de la fréquence).

J'ai donc essayé de corriger çà. Mais mes calculs ne sont satisfaisants que sur 2 octaves (celles dans la majeure partie des partitions suffisent). Si quelqu'un a mieux, je serais ravi d'en avoir connaissance.

* Les lignes 300 à 370 permettent la correction des notes, 400 à 415 celle de temps.

* Le reste jusqu'à 500 sauvegarde les données obtenues.

* Le sous-programme 2000 est le même qui permet de choisir entre l'écriture et l'écoute.

* Et le sous programme 50.000 donnant les explications nécessaires (déjà dit).

P.S. : Les non-possesseurs de CE-150 remplaceront bien sûr tous les LPRINT par PRINT et devront s'acheter une feuille de papier et un bon stylo.

```
1:CLEAR :DIM F(1
  00),D(100):A$=
  "60607F0363637
  F"
5:WAIT 0:GPRINT
  A$:GCURSOR 21:
  PRINT "Program
  mes Musicaux":
  GCURSOR 148:
  WAIT 100:
  GPRINT A$:GOTO
  2000
10:WAIT 10:PRINT
  "Impression de
  s codes (O/N)"
20:GOSUB 900
30:IF R$="O"GOSUB
  50000
40:INPUT "TEMPO D
  E LA NOIRE = "
  ;T
45:I=I+1:C=0
50:PRINT "NOTE No
```

```
     .";I;" = "
  60:R$=INKEY$
  70:IF R$="D"LET N
     =0:GOTO 90
  71:IF R$="R"LET N
     =2:GOTO 90
  72:IF R$="M"LET N
     =4:GOTO 90
  73:IF R$="F"LET N
     =5:GOTO 90
  74:IF R$="L"LET N
     =9:GOTO 90
  75:IF R$="S"PRINT
     "S":GOTO 80
  77:GOTO 60
  80:R$=INKEY$
  82:IF R$="O"LET N
     =7:GOTO 90
  83:IF R$="I"LET N
     =11:GOTO 90
  85:GOTO 80
  90:PRINT "ALTERAT
     ION = "
  92:R$=INKEY$ :IF
     (R$<>" ")*(R$<
     >"B")*(R$<>"D"
     )GOTO 92
  95:IF R$="D"LET N
     =N+1
  97:IF R$="B"LET N
     =N-1
 100:PRINT "OCTAVE
     = "
 110:R$=INKEY$ :IF
     (R$<>"H")*(R$<
     >"B")GOTO 110
 120:IF R$="H"LET N
     =N+12
 130:PRINT "RYTHME
     = "
 140:R$=INKEY$ :IF
     (ASC R$<48)+(
     ASC R$>57)GOTO
     140
 150:GOSUB (1000+
     VAL R$)
 160:PRINT "POINT =
     "
 170:R$=INKEY$ :IF
     (ASC R$<48)+(
     ASC R$>50)GOTO
     170
 180:IF R$="1"LET V
     =V*1.5
 190:IF R$="2"LET V
     =V*1.75
 200:F(I)=215*EXP (
     -.06*N):F(I)=
     INT F(I)
 210:D(I)=(12000000
     /T)/(3.97*F(I)
     +.5)*V:D(I)=
     INT D(I)
 215:IF C=1LET C=0:
     GOTO 330
 220:BEEP 1,F(I),D(
     I)
 230:PRINT "D ACCOR
     D ?"
 240:GOSUB 900
 250:IF R$="N"GOTO
     50
 260:PRINT "ENCORE
     ? "
 270:GOSUB 900
 280:IF R$="O"GOTO
     45
 285:J=I
 290:FOR I=1TO J:
     BEEP 1,F(I),D(
     I):NEXT I
 300:PRINT "SATISFA
     IT ? "
 310:GOSUB 900
 320:IF R$="O"GOTO
     400
 325:CLS :BEEP 2:
     CURSOR 8:WAIT
     80:PRINT "Corr
     ection":WAIT 1
     0:I=1
 330:BEEP 1,F(I),D(
     I):PRINT "CORR
     ECT ?":GOSUB 9
     00
 335:CLS
 340:IF R$="O"GOTO
     360
 350:C=1:GOSUB 50
 360:I=I+1:IF I>J
     GOTO 290
 370:GOTO 330
 400:PRINT "TEMPO C
     ORRECT ?":
     GOSUB 900:IF R
     $="O"GOTO 420
 410:WAIT 100:PRINT
     "ANCIEN TEMPO
     =";T:INPUT "NO
     UVEAU TEMPO ="
     ;U
 415:FOR I=1TO J:D(
     I)=D(I)*T/U:
     NEXT I:FOR I=1
     TO J:BEEP 1,F(
     I),D(I):NEXT I
     :T=U:GOTO 400
 420:INPUT "QUEL ES
     T LE TITRE ?",
     T$:IF LEN T$>1
     6BEEP 3:PRINT
     "TROP LONG !":
     GOTO 420
 425:WAIT 10:PRINT
     "Dois-je enreg
     istrer ?":
     GOSUB 900:IF R
     $="N"GOTO 460
 430:PRINT "Magneto
     phone pret ?":
     GOSUB 900:IF R
     $="N"GOTO 430
 440:PRINT #T$;T$,F
     (*),D(*),J
 460:PRINT "Dois-je
      imprimer la l
     iste?":GOSUB 9
     00:IF R$="N"
     END
 470:CSIZE 1:COLOR
     0:LPRINT "
     DONNEES PERMET
     TANT DE JOUER:
     ":COLOR 2:
     CSIZE 2:LF 1
 480:LPRINT TAB ((1
     8-LEN T$)/2);T
     $:COLOR 0:
     CSIZE 1
 490:FOR I=1TO J:
     LPRINT F(I);".
     ";D(I);".";:
     NEXT I
 500:TEXT :LF 5:END
 600:PRINT "Tout es
     t pret ?":
     GOSUB 900:IF R
     $="N"GOTO 600
 610:INPUT #T$;T$,F
     (*),D(*),J
 620:FOR I=1TO J:
     BEEP 1,F(I),D(
     I):NEXT I
 630:PRINT "Voulez-
     vous la liste
     ?":GOSUB 900:
     IF R$="O"GOTO
     470
 640:PRINT "Dois-je
      le rejouer ?"
     :GOSUB 900:IF
     R$="O"GOTO 620
 650:END
 900:R$=INKEY$ :IF
     (R$<>"O")*(R$<
     >"N")GOTO 900
1000:V=8:RETURN
1001:V=4:RETURN
1002:V=2:RETURN
1003:V=1:RETURN
1004:V=.66:RETURN
1005:V=.50:RETURN
1006:V=.33:RETURN
1007:V=.25:RETURN
1008:V=.166:
     RETURN
1009:V=.125:
     RETURN
2000:CLS :WAIT 10
     0:PRINT "Vou
     lez-vous :":
     PRINT "Recop
     ier une part
     ition ou"
2010:PRINT "jouer
      un air enre
     gistre ?":
     WAIT 10:
     PRINT "recop
     ier=1 jouer=
     2"
2020:R$=INKEY$ :
     IF R$="1"
     GOTO 10
2030:IF R$="2"
     GOTO 600
2040:GOTO 2020
50000:TEXT :COLOR
     0:LPRINT TAB
     4;"Codes pou
     r":LPRINT "e
     criture  mus
     icale":
```

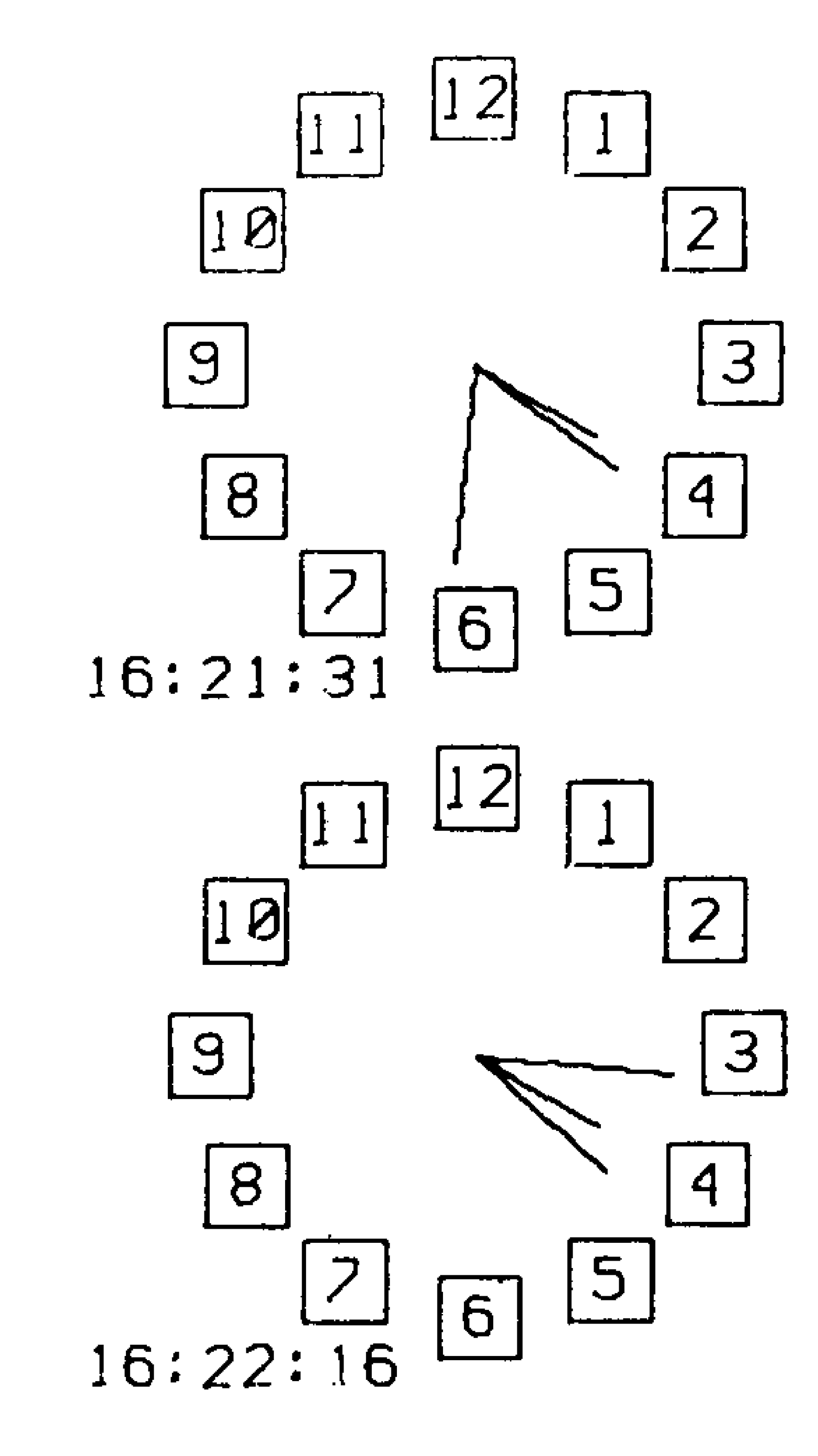
```
         LPRINT TAB 3
         ;"sur  PC-15
         00"
  50001:GRAPH :COLOR
         2:LINE (60,0
         )-(150,0):
         TEXT :LF 2:
         COLOR 0
  50002:CSIZE 3:
         LPRINT "Note
         s:":CSIZE 2:
         LPRINT " DO=
         D  RE=R  MI=
         M":LPRINT "
         FA=F SOL=SO
         LA=L"
  50003:LPRINT TAB 7
         ;"SI=SI":LF
         1
  50005:CSIZE 3:
         LPRINT "Alte
         rations:":
         CSIZE 2:
         LPRINT TAB 2
         ;"Becarre=SP
         ACE":LPRINT
         TAB 2;"  Die
  50006:LPRINT TAB 2
         ;"   Bemol=B"
  50010.LF 1:CSIZE 3
         LPRINT "Oct
         aves:":CSIZE
         2:LPRINT TAB
         4;"Basse=B":
         LPRINT TAB 4
         ;"Haute=H":
         LF 1
  50015:CSIZE 3:
         LPRINT "Ryth
         mes:":CSIZE
         2:LPRINT " C
         arree=0 Rond
         e=1":LPRINT
         "Blanche=2 N
         oire=3"
  50016:LPRINT " Tri
         olet noires=
         4":LPRINT
         TAB 5;"Croch
         e=5":LPRINT
         "Triolet cro
         ches=6"
  50017:LPRINT "Doub
         le cr=7 Sext
         =8":LPRINT "
           Triple cro
         che=9"
  50020:LF 1:CSIZE 3
         :LPRINT "Poi
         nts:":CSIZE
         2:LPRINT "In
         diquer 0 1 o
         u 2."
  50025:GRAPH :COLOR
         2:LINE (60,0
         )-(150,0):
         LINE (80,-2)
         -(130,-2):
         TEXT :LF 5
  50030:RETURN
```

Graphismes

Voici des petits programmes graphiques amusants : Immortalisez l'heure en l'imprimant graphiquement.
A ma Sharp à quartz aiguille il était...

I.S.

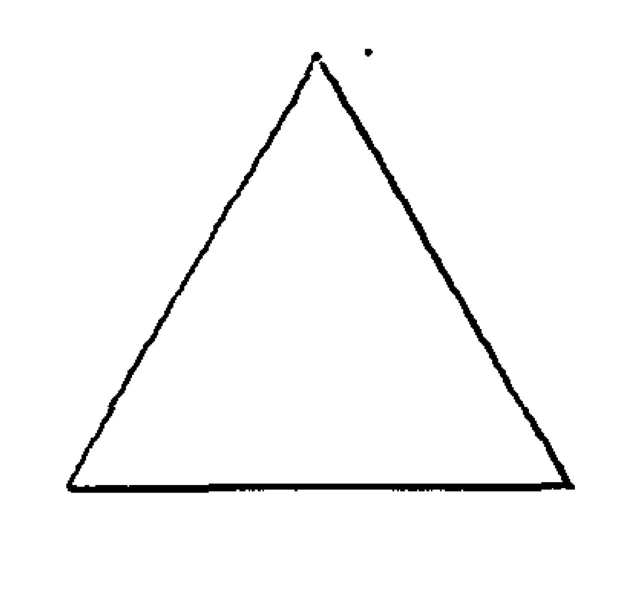
Horloge

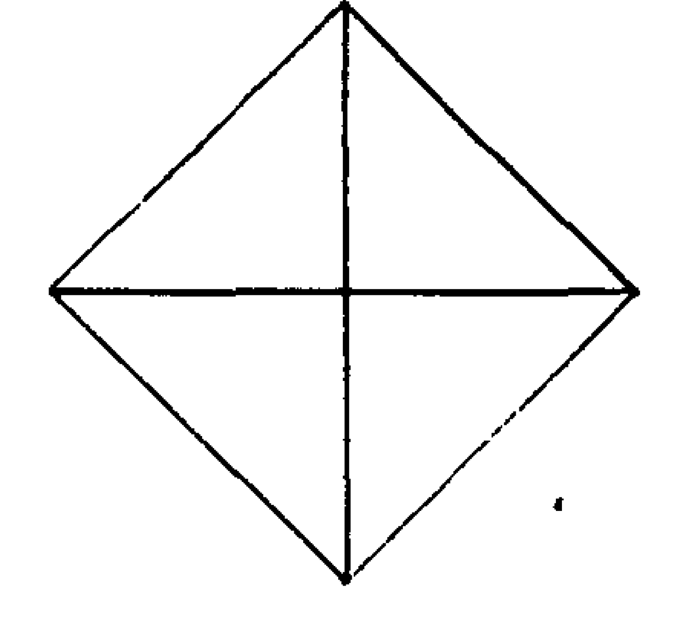


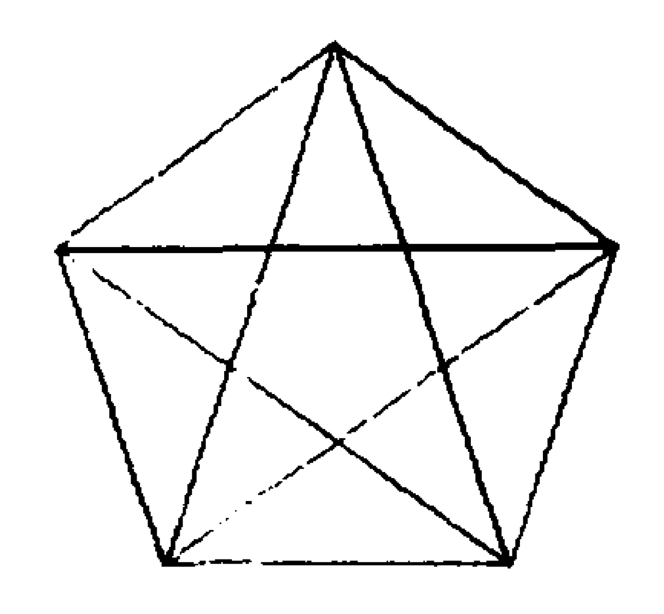

```
300:"H"GRAPH :
     GLCURSOR (105,
     0):D=0:SORGN :
     FOR I=30TO 360
     STEP 30:D=D+1:
     G=4:IF D>9LET
     G=10
310:A=SIN I*80:B=
     COS I*80:LINE
     (A,B)-(A+25,B+
     25),,,B:
     GLCURSOR (A-G,
     B+7):LPRINT D:
     NEXT I:
     GLCURSOR (12,1
     2):SORGN
320:T=TIME *100:S=
     (T-INT T)*100:
     T=INT T/100:M=
     (T-INT T)*100:
     T=INT T/100:H=
     (T-INT T)*100
330:LINE (0,0)-(
     SIN (360*H/12)
     *45,COS (360*H
     /12)*45)
340:LINE (0,0)-(
     SIN (360*M/60)
     *55,COS (360*M
     /60)*55)
350:LINE (0,0)-(
     SIN (360*S/60)
     *61,COS (360*S
     /60)*61)
360:GLCURSOR (-100
     ,-100):TEXT :H
     $=RIGHT$ ("00"
     +STR$ H,2):M$=
     RIGHT$ ("00"+
     STR$ M,2):S$=
     RIGHT$ ("00"+
     STR$ S,2)
370:LPRINT H$;":";
     M$;":";S$:END
```

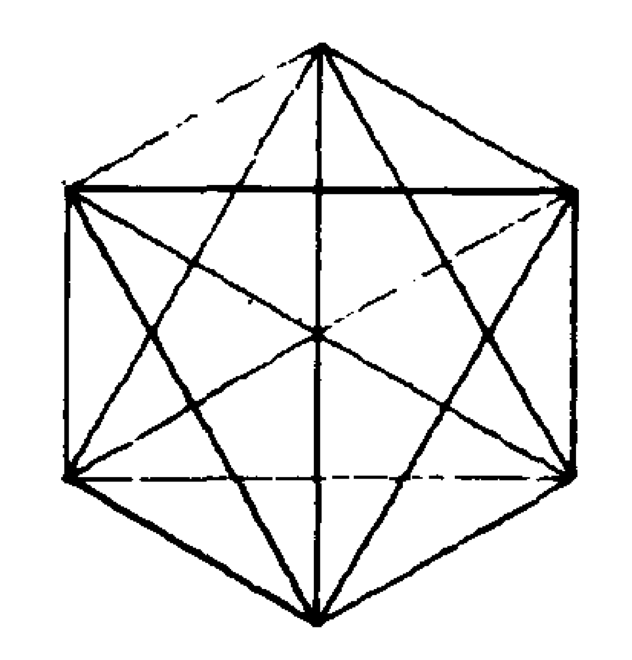
N'enviez plus les rosaces de nos belles cathédrales gothiques, créez les vôtres du triangle à la rosace la plus complexe.

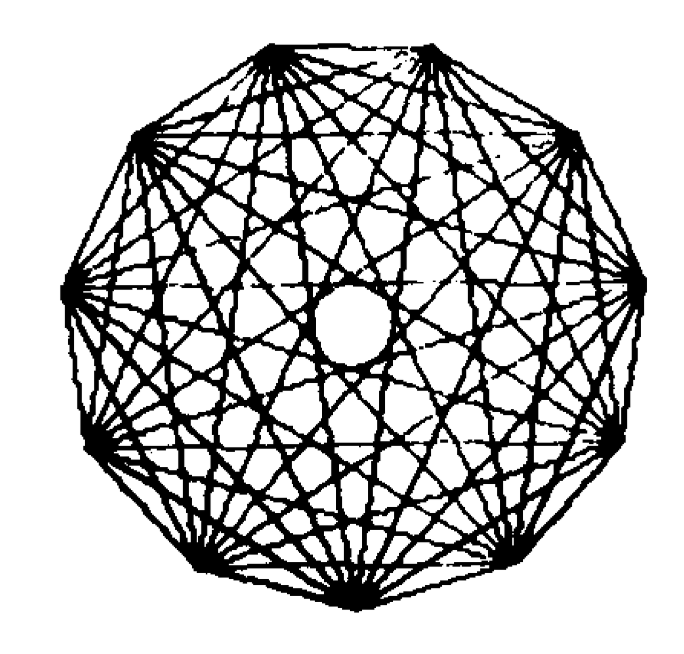
I.S.





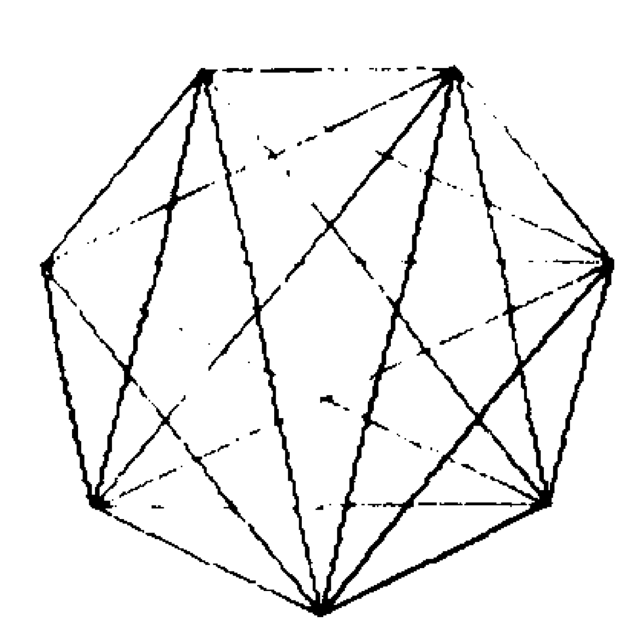


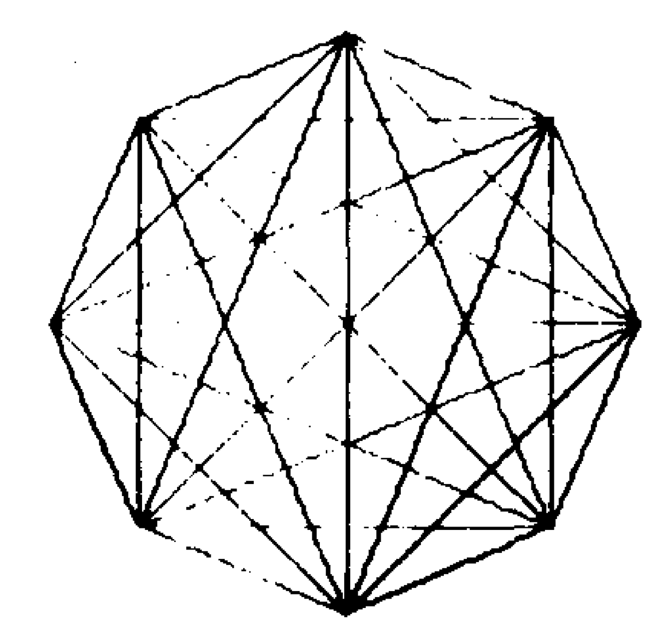


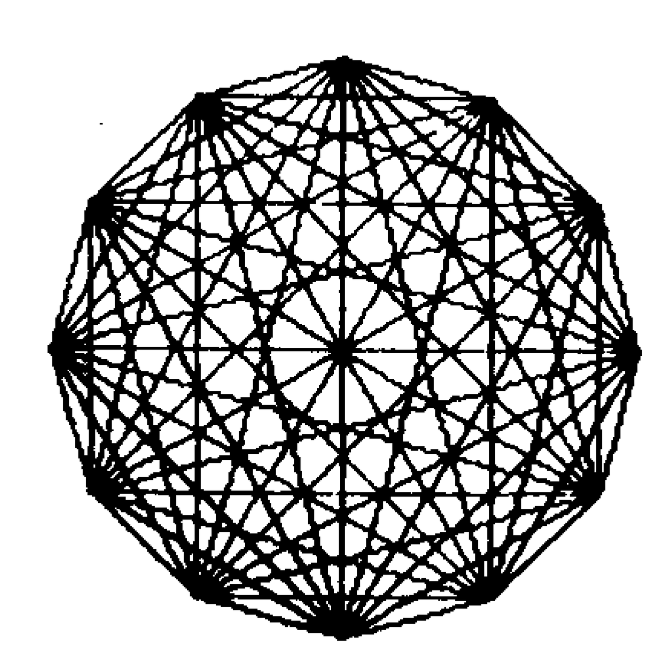


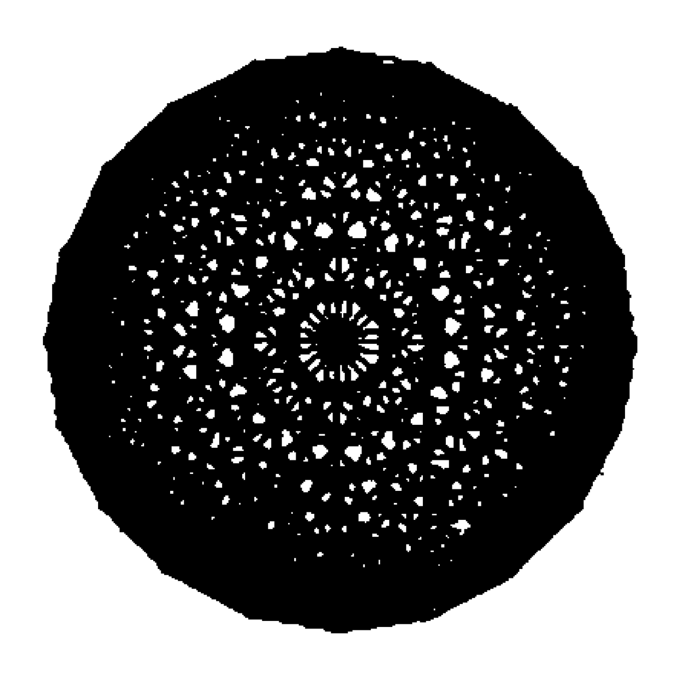
```
10:GRAPH :
    GLCURSOR (105,
    0):SORGN :
    INPUT "X?";X:
    DIM A(X),B(X):
    C=0:FOR I=0TO
    360STEP 360/X
```

```
20:I=I+360/X-INT
   (360/X):A(C)=
   SIN I*80:B(C)=
   COS I*80:C=C+1
   :NEXT I:C=C-2
30:FOR I=0TO C-1:
   FOR J=I+1TO C:
   LINE (A(I),B(I
   ))-(A(J),B(J))
   :NEXT J:NEXT I
   :END
```

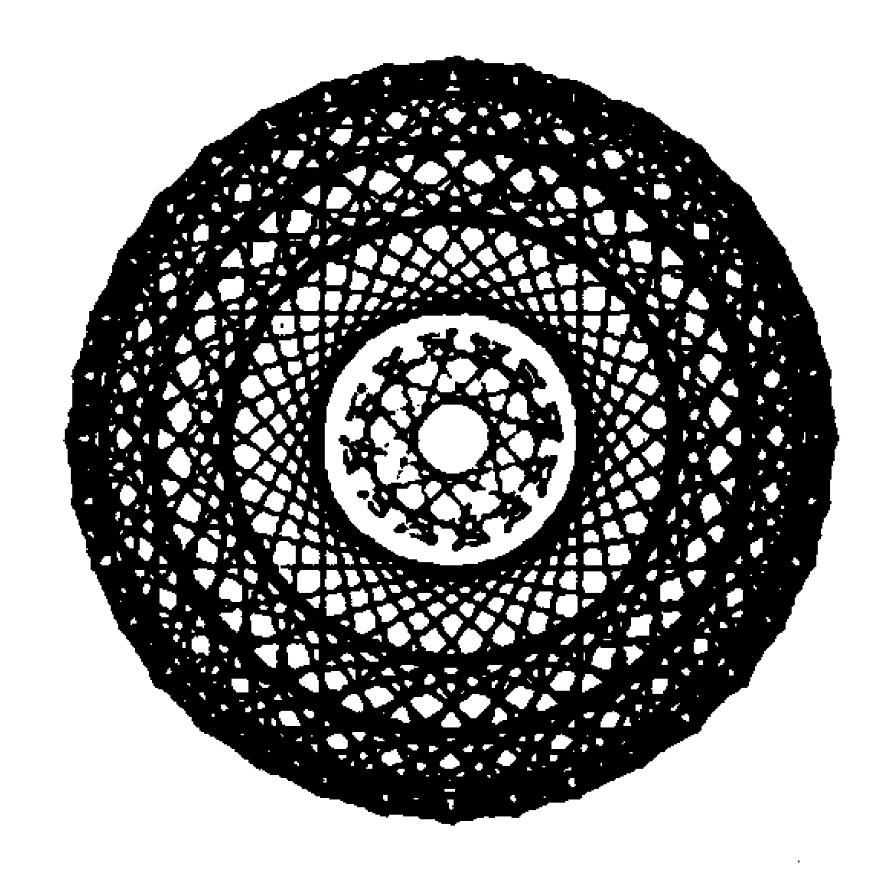








Macaron



```
5:GRAPH :
  GLCURSOR (107,
  -150):SORGN :
  COLOR 3
10:FOR I=0TO 360
   STEP 10:FOR J=
   20TO 120STEP 3
   0:LINE (COS I*
   105,SIN I*105)
   -(COS (I+J)*10
   5,SIN (I+J)*10
   5)
20:NEXT J:NEXT I:
   COLOR 2
30:FOR I=0TO 360
   STEP 30:FOR J=
   20TO 120STEP 6
   0:LINE (COS I*
   30,SIN I*30)-(
   COS (I+J)*30,
   SIN (I+J)*30)
40:NEXT J:NEXT I: END
```

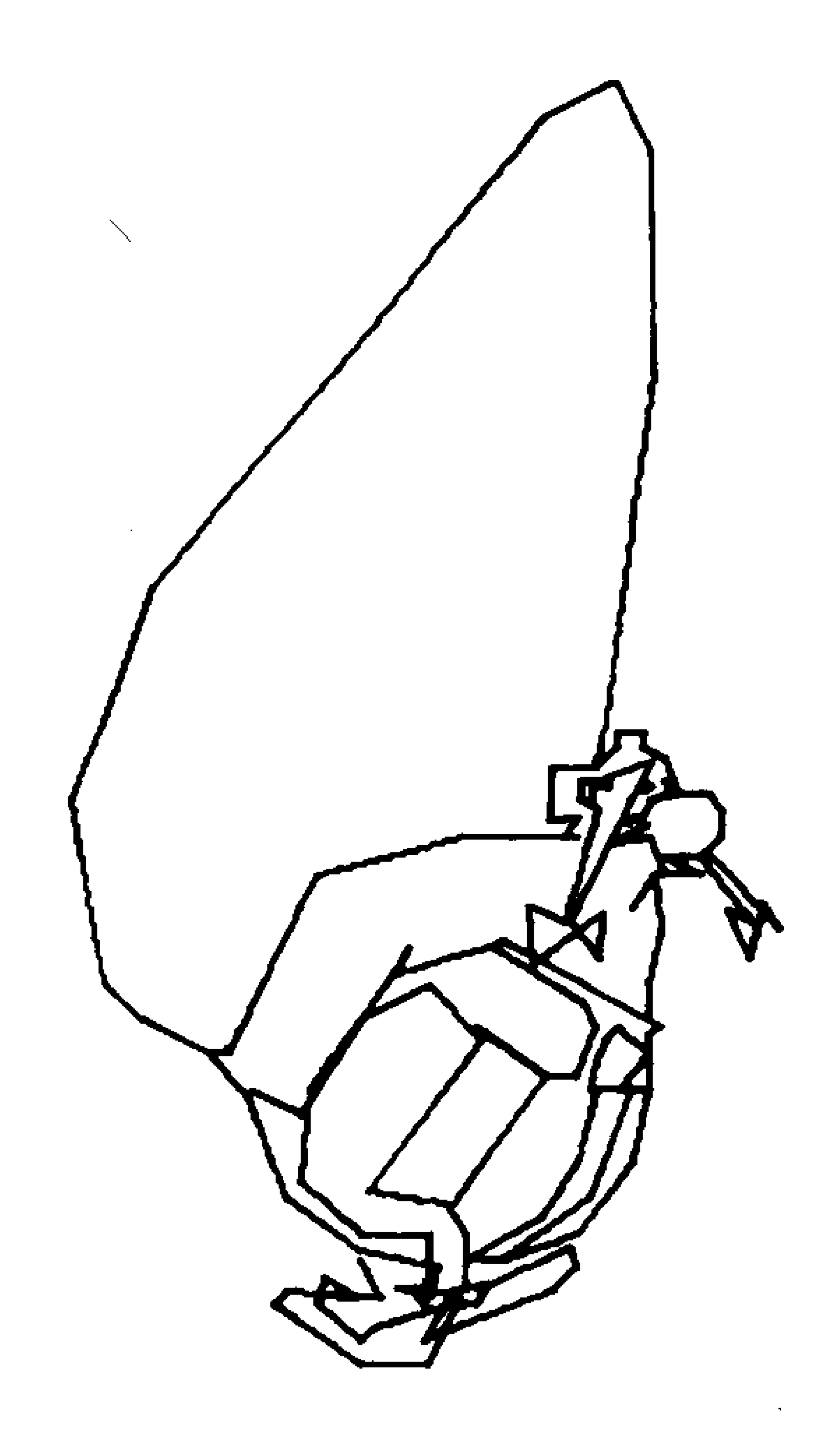
Grégory Guida.

Programme surprise.

```
  1:DATA 0,0,23,24,25,22,26,24,27,24
    ,27,21,34,20,40,17,45,16,45,17,5
    0,10,55
  2:DATA 16,54,14,57,13,57,11,60,13,
    62,11,60,13,67,12,66,20,69,24,71
    ,22,76,26,77,25
  3:DATA 80,25,81,23,87,24,90,21,95,
    21,99,23,99,25,103,29,107,31,107
    ,32,109,35,107,37
  4:DATA 104,35,100,37,101,43,97,46,
    90,48,101,40,102,51,90,54,98,56,
    101,50,100,59,97
  5:DATA 61,97,63,93,61,93,60,92,58,
    90,60,95,70,99,75,97,77,99,79,10
    1,76,103,70,104
  6:DATA 90,107,96,102,96,100,90,95,
    97,92,100,80,102,87,100,84,100,8
    3,102,80,104,79
  7:DATA 100,70,109,75,105,73,107,73
    ,110,65,114,64,117,62,116,60,120
    ,52,116,52,109,54
  8:DATA 109,50,100,50,105,42,104,40
    ,100,43,99,39,90,32,99,30,98,29,
    100,31,103,27,102
  9:DATA 24,104,25,97,27,96,27,92,29
    ,89,26,80,25,91,24,91,22,87,21,9
    0,16,80,15,93
 10:DATA 11,93,10,90,0,92,3,90,1,90,
    1,87,4,88,5,86,2,87,1,85,5,84,1,
    84,2,82,3,80,5
 11:DATA 80,6,83,15,70,16,75,19,76,1
    0,73,19,72,23,74,25,72,21,72,21,
    71,23,69,21,67
 12:DATA 26,62,30,60,29,59,31,56,28,
    54,32,52,35,46,35,43,32,46,29,51
    ,28,49,28,41
 13:DATA 30,42,31,40,28,39,26,20,23,
    24,9999,9999
 14:DATA 4,42,99,44,99,45,100,47,100
    ,46,98,49,99,48,99,51,98,52,95,5
    6,93,60,92,63
 15:DATA 86,71,89,73,86,76,83,70,75,
    9999,9999,25,72,27,73,30,75,36,7
    5,40,72,44,75
 16:DATA 50,76,51,70,55,80,59,70,63,
    78,65,73,65,70,60,68,72,63,75,59
    ,74,56,76,51,74
 17:DATA 47,73,44,75,41,76,42,9999,9
    999,34,43,35,40,30,39,40,30,41,3
    6,44,33,49,32
 18:DATA 50,27,40,22,43,21,45,17,43,
    13,9999,9999,80,20,81,26,79,26,7
    9,31,82,33,79
 19:DATA 34,70,30,80,45,79,50,80,55,
    83,56,84,55,87,57,87,53,89,50,90
    ,60,92,50,95,60
 20:DATA 95,62,97,63,102,62,105,63,1
    02,62,97,63,97,65,93,65,90,60,99
    99,9999
100:READ A:READ B
110:CSIZE 2:LF 10:J=0:GRAPH :SORGN
115:INPUT "Echelle?";E
117:GOSUB "Q"
118:GOTO 200
120:"Q":READ X:READ Y
130:READ V:READ W:IF V=9999OR W=9999
    THEN RETURN
140:LINE (X*E,Y*E)-(V*E,W*E),0,J
145:WAIT 0:PRINT V,W
150:X=V:Y=W:GOTO 130
200:READ C:J=2:FOR I=1TO C
210:GOSUB "Q"
220:NEXT I:LINE (130,4)-(130,0)-(130
    +(41*E/1.95),0)-(130+(41*E/1.95)
    ,3):CSIZE 1:LPRINT " 200 KM
```

Grégory Guida.

Obélix



```
 10:LF 17:GRAPH
 20:RESTORE
 25:FOR A=1TO 11
 30:READ X:READ Y:GLCURSOR (X,Y)
 40:READ X:READ Y:IF X=999GOTO 90
 50:LINE -(X,Y):GOTO 40
 90:NEXT A:END
100:DATA 157,104,163,220,174,304,174
    ,369,163,391
101:DATA 141,300,22,239,0,174,11,119
    ,22,100,43,90
102:DATA 49,98,74,152,119,163,152,16
    3,140,163,152,169
103:DATA 143,160,143,184,157,184,183
    ,190,163,195,171,195
104:DATA 171,190,170,187,182,176,174
    ,174,182,176,191,176
105:DATA 195,171,195,163,189,157,206
    ,136,197,141,204,126
106:DATA 200,143,213,135,190,159,189
    ,157,170,157
107:DATA 189,152,170,152,174,163,163
    ,161,174,167,999,999
108:DATA 43,98,43,97,54,85,65,54,87,
    38,111,33,100,24
109:DATA 98,27,100,27,100,43,87,43,6
    9,59,72,87,87,100
110:DATA 100,119,124,102,89,56,113,5
    2,119,43,119,26
111:DATA 104,24,100,20,122,22,126,20
    ,115,26,106,11
112:DATA 145,26,152,33,150,39,122,26
    ,999,999
113:DATA 115,52,143,91,122,106,143,9
    1,152,91,150,106,154,113
114:DATA 126,130,130,132,170,106,174
    ,104,174,87,156,87
115:DATA 158,95,161,102,165,106,174,
    98,165,87,158,87
116:DATA 152,65,141,48,119,35,130,35
    ,156,54,169,87
117:DATA 174,87,169,65,152,43,130,35
    ,999,999
118:DATA 126,130,90,122,102,130,87,1
    00,69,70,54,87,999,999
119:DATA 87,35,93,24,85,24,76,30,74,
    22,87,11,91,15,100,20,999,999
120:DATA 85,24,65,26,61,22,87,4,100,
    4,117,24,999,999
121:DATA 156,174,156,170,161,170,161
    ,180,152,180,152,174,150,171
122:DATA 140,137,137,142,139,124,160
    ,140,159,126,150,137
123:DATA 174,186,150,180,999,999,169
    ,141,176,152
124:DATA 170,135,174,119,174,100,999
    ,999,171,100,174,170,171,176,999
    ,999
125:DATA 170,100,100,179,170,176,999
    ,999,167,167,169,169,171,167,174
    ,174,999,999
```

Christophe Camus.

Courrier

Lors de l'exécution de l'un des programmes précédemment publié dans votre bulletin, une erreur est survenue : le curseur clignotait sur « A(N) ». Pourriez-vous m'expliquer cette erreur ?

M. R. Piron — Villemonble

Vous savez que la mémoire est repartie entre variables et lignes basic de programme. Ainsi. plus votre programme basic sera important, plus le nombre de variables diminuera. Ainsi, sachez que de A(n) sera valable pour les N inférieurs au nombre de « mémories » affiché après un MEM ajouté aux 26 variables de A à Z.

Etant possesseur d'un PC 1211, j'ai récemment acheté une PC-1500. J'ai donc été amené à transcrire ma logithèque sur PC-1211 sur ma nouvelle « merveille ». Cependant, la PC-1211 possédait la particularité de pouvoir appeler les mémoires de A à Z par les expressions A(1) à Z (26).

Cette même manipulation n'est pas possible sur mon 1500 (il faut utiliser DIM, et de plus, on ne récupère pas dans ce tableau ce que l'on avait chargé de A à Z). Comment dois-je faire pour transcrire ce programme ?

M. J. Antoine

Mais si, l'équivalent existe ! Certes, redimensionner le tableau A ne permet pas de récupérer ce que vous avez stocké de A à Z, mais comme cela est indiqué (d'accord, çà n'est pas très développé, mais c'est écrit !), vous pouvez appeler les mémoires de A à Z et A $ à Z$ par G (1) à G 26 et à G S (1) à G S (26).

Ayant récemment acheté une PC 1500, accompagnée de l'imprimante couleur, j'aurais quelques remarques à formuler : le papier d'origine était d'excellente qualité alors que les rouleaux achetés par la suite présentaient de gros défauts, ils étaient trop courts et recouverts de bandes rouges !!

Comment remédier à cela ?

Nous avons en effet remarqué les défauts des rouleaux achetés sur le marché français. Actuellement SHARP est en tractation pour acheter au Japon les rouleaux pour la CE-155. Nous espérons ainsi retrouver dans le papier de remplacement la qualité du papier d'origine.

N'ayant pas les moyens de me procurer un nouveau magnétophone muni (sic !) d'une prise de télécommande, j'ai opéré sans elle ; après avoir entré SAVE, j'ai attendu durant plus de 20 minutes l'arrêt du beep...Sans résultats HELP ME PLEASE !!!...

Vassalo Thierry - Montreuil

La prise de télécommande que ne possède pas votre magnétophone à pour usage de stopper l'enregistrement dès que celui-ci est terminé. Si vous ne pouvez en profiter, il vous faudra stopper votre magnétophone manuellement (le compteur de votre appareil vous sera très utile pour le repérage).

J'ai essayé d'utiliser le « truc » permettant l'accès à la rotation hexadécimale du PC-1211 (Ligne de Basic que l'on retrouve dans les variables A (204)...) Cependant, même si, après un NEW, je retrouve dans A (204) toutes sortes de caractères inattendus (%, Y...) ma PC 1211 reste insensible à toutes les manipulations décrites par ce truc. Pourriez-vous m'éclairer ?

J.J. Guittoux, Paris.

Le « truc » dont vous nous parlez n'est efficace que pour les PC-1211 de la « première génération ». En effet, le défaut que cette astuce exploite a été supprimé pour les machines plus récentes. Cependant, il existe une autre possibilité. Enregistrez en PRO un programme sauvegarde en RESERVE, le résultat vous étonnera...

Sharpentiers, Sharpentiers

Je possède une PC-1211, ainsi que l'imprimante. Je désirerais étendre mon système, notamment avec une interface VIDEO.

Une telle extension existe-t-elle ?

M. P. Dusard

Vous avez été nombreux à nous poser cette question !

Voyez l'éditorial.

Editorial 1211

Un oubli s'était glissé dans le dernier bulletin, un auteur n'avait pas été mentionné, alors Monsieur DELCOURT veuillez accepter une nouvelle fois nos excuses et encore merci pour votre programme de calcul de bulletin de salaire.

Une bonne nouvelle, un particulier, Monsieur Pierre AUSPITZ a décidé de concevoir un interface vidéo pour la PC 1211. Il aimerait savoir si vous êtes vraiment nombreux à être intéressés par ceci. Si oui, écrivez-nous. Plus vous serez nombreux moins cela coûtera cher, et une fabrication sera peut-être envisagée.

Dans la série face cachée du PC 1211 de nouvelles astuces pour utiliser en programmation la face cachée de la 1211. C'est une application intéressante qui vous permet d'espérer que vous trouverez bientôt d'autres trucs.

En attendant voilà un bulletin 1211 bien mince ce trimestre.

A bientôt.

Luc BURELLER.

Je m'appelle Jean-Cyril LAURENT, je fais partie du Club PC 1211 depuis Décembre 81, date à laquelle j'ai acheté mon premier ordinateur : le SHARP PC 1211.

Depuis ce temps, tout ce qui touche cette machine me passionne. Aussi le club et son bulletin représentent beaucoup pour moi

D'ailleurs, j'ai approfondi les "ASTUCES" que vous communiquiez dans le n° 3 du bulletin = l'apparition du signe curseur "-".

Tout d'abord, j'ai trouvé un autre procédé pour l'obtenir car sur ma machine, celui que Monsieur ZUMMER nous donne ne marche pas.

Alors en combinant tout ce que je savais sur les "dessous du PC-1211", j'ai obtenu :10 opérations pour la création de ligne par programme. (C'est un peu compliqué mais absolument efficace).

1— Faire NEW

2— Affecter à la variable A (203) la valeur « habituelle » A (203) = 0,021052101

3— En mode PRO inscrire : 10 PRINT (ou n'importe quoi d'autre, mais de dépassant pas 1 ou 2 pas). 20 END.

4— Efacer la ligne 20 (20, ENTER).

5— Puis en mode RUN, démarrer le programme approprié par l'instruction.

6— Après l'affichage de π, taper MEM, ENTER. La mémoire restante apparaît.

7— Appuyer 1 fois sur ON.

8— Taper sur ENTER ou faire CONT, une erreur apparaît (bizarre).

9— Retourner en mode PRO après avoir appuyé sur CL et appuyer sur une des touches de défilement de lignes de programme ↓ ou ↑.

La ligne 20 réapparaît malgré le fait que l'on ait effacé celle-ci précédemment.

10— Enfin, appuyer sur la touche ↓ et la ligne 10 «—» apparaît. Ensuite, vous savez ce qu'il reste à faire... (, ENTER).

Voici donc comment j'obtiens le signe curseur «—».

Maintenant, mes découvertes avec ce signe « Hors-Clavier » ; il constitue une mémoire fixe.

Vous pouvez, par exemple, vous en servir comme variable telle que A,B ou A (150), mais aussi de compteur (mais c'est compliqué car dans une même ligne de programme il me semble impossible d'obtenir deux fois le signe curseur «—»), ainsi on peut trouver dans un programme :

- 10 : INPUT —
- 20 : PRINT —
- 30 : FOR = 1 to 10. —
- 40 : PAUSE
- 50 : NEXT —

«—» fait donc partie de la famille des mémoires fixes, il n'entre pas dans la mémoire souple dite de programme. (Très utile lorsque l'on a besoin d'une variable différente de ces A (...) trop encombrants !)

Notez que l'on peut mettre en pratique les 10 instructions précédentes avec tout un programme en mémoire en procédant ainsi :

Il faut donner à la deuxième mémoire souple accessible la valeur 0,0221052101 (en demandant « MEM » et en ajoutant 25 au nombre de « MEMORIES » restant).

Ensuite, et cela explique le 25, taper :

998= PRINT

999= END, effacer cette ligne, faire RUN, 998, etc.

Un peu plus difficiele à obtenir, quelquefois on obtient : 12 «—».

Bref, à partir de cela, on peut considérer que le PC-1211 possède 205 mémoires accessibles...

J.C. LAURENT

Logithèque 1211

de M. BALMETTRE Didier
Attaque nucléaire — jeux
L'île au trésor — jeux
Q.I.

de M. BUSCHINI Philippe
Calcul de fréquence (Thomson)
Impédance d'un circuit résonnant série.

de M. X
Ordinateur de bord.
Le château de TIHP — jeux
PRINT — utilitaire
DCA — jeux

de M. DELYS Jean Robert
Erreur n'est pas comptée. Comparaison de prix utile aux ménagères.

de M. COTILLON
Calcul d'élément déprimogènes.
Calcul de soupapes de sûreté.

de M. MORTIER
Contrôle des CA 12

BALISTI de M. FICHANT
Calcul des dérivés transversales et verticales d'une balle soumise au vent

FILTRES de M. Bruart
Calcul de l'ordre d'un filtre

PPCMGCD de M. VESIGNOT
Calcul du PPCM et du PGCDPREMIER

de M. VESIGNOT
Calcul du nombre premier

PT MER de M. CHOSSELER
Calcul des coordonnées géographiques

POEME de M. VANDEPUTTE
Création aléatoire de phrases

BIORYTHM de M. STERMANN

SHANNON de M. POINSOT
Calcul de diversité de SHANNON

MASTER MIND de I.S.
Jeux
AGENDA de I.S.
Plannifier les 26 jours à venir

***SIMPLI* de I.S.**
Simplification de fractions et recherche de fractions à partir d'un nombre

SECTOR de I.S.
Jeux

FACTO de M. PAGLI
Méthode du STIRLING

FIX V.1 de I.S.
Arrondi

IMPOTS de M. THIBAUT
Impôts 81

PUISSANCE de I.S.

BALL GAME de I.S.
Jeux

CONVERSION ANGLOSAXONNE de I.S.
Conversion des principales mesures anglo-saxonnes (livre, mile, gallon, fahrenheit)

FACTORIEL de I.S.
(mantisse complète, très long à l'exécution)

CALCULS PRECIS de M. VESIGNOT
Multiplication, division, de nombres à plus de 44 chiffres significatifs

de M. J.F. CHAPIN
NOTEXMO : Notes externes et moyennes utiles aux professeurs pour leurs conseils de classes.

WEIGHT : Programme de conversion anglosaxonne sur les poids.

RELATIF : Emploi des pronoms relatifs en anglais

MCQ REL : Questionnaire sur les pronoms relatifs en anglais
SUPERVI : Verbes irréguliers anglais

TENSE : Test sur l'emploi des temps en anglais

MEASURE : (en 3 parties) Conversion de distance, change, température, longueur, volume.

de M. BOISRAYON
Cœf BAL : Calcul du cœfficient balistique d'une balle de petit calibre

BALARM : Balistiques des armes rayées portatives

FACTO : Factorielle formule de Stirling

APPROX : Fraction approximant un réel R

IMPLICI : Résolution d'une fonction implicite

WEDINT : Intégration par la méthode Deweddle

RACCAR : Extraction d'une racine carrée entrée par la méthode arithmétique

de M. CHAMPAGNE
BANQUE : Gestion d'au plus 25 crédits ou débits

de M. BODET
INVADERS : Jeu

BAC : Intéressant de savoir si vous l'aurez, en fonction de vos notes actuelles.

de M. DARJO
CALENDRIER : Imprime le calendrier d'une année

SIMON : Jeu, mémorisation de nombres

de M. COTILLON
SOUPAPE : Calculs de soupapes

de M. MULLOT
SERVAC : Service achats, commandes sur 5 mois

de M. CLOUE
FONC : Traceur de fonctions (nécessite CE 122)

de M. BARRE
TRI : Méthode de Shell : 70 données

de M. GAY
MATRICE : Calcul sur la matrice d'ordre X (X 8)

FONCT : Recherche d'une fonction vérifiant des donnés

de M. TAUZIEDE
DISTAN : Calcul d'une distance réelle sur une planète

de M. MADEUF
POLY : Calculs de polynômes

de M. NOEL
TOPOGRA : Relèvement italien, méthode de POTHENDT

de M. HAL BARONN
TUYAU : Calcul de la perte de charge d'une tuyauterie l'air de secteur circulaire

de M. ZUMMER
POLIDIR : Division des polynômes

de M. BRENAS
CLEF DE RIB

de M. PIACENTINO
Jeu de la boîte noire.

de M. ZUMMER
Naval 1, Naval 2 : Jeu de bataille navale
Bourse : Jeu de bourse

de M. BALMETTE Didier
FLECH : Jeu de fléchettes

de M. RAMPAL Guy
SUBROG : Calcul automatique de la tarification de la facture subrogatoire de la Sécurité Sociale

de M. BODET Pierre
TENNIS

MAX : Jeu des chiffres et des lettres

NAVAL : Jeu de la Bataille navale (contre PC 1211)

de M. VALLY Bruno
QI : Calcul du QI
PI : Aptitude verbale

de M. BOUTIN G.
ASTRO : Astronomie : calculs des équinoxes en coordonnées équatoriales

de M. GRESSARD J.-M
TIERCE

de M. GAY Thierry
JACKPOT

de M. AIUTI Laurent
STATIS : Statistiques

de M. CHAPIN J.-F
CLASS 82 : Gestion des élèves d'une classe

CIRCLAS

SELCLAS SECL 2 : Utiles à tous les professeurs

FICH DAT

STAT DAT

AGEFICH

de M. SANTARELLI Eric
JEU DE L'OIE

JEU DU ROBOT FOU

PENDU : Jusqu'à 24 lettres

de M. DELCOURT Alain
ISOLEMENT

PROFILES

PAYE

de M. MORTIER
CA12 : Destiné aux petits contribuables soumis au régime simplifié de la T.V.A. (Impôts).

de M. CHAPIN
Jury bac
Essay
Delib-d (variante Jury bac)

de M. J.L. BECHENNEC
Astro 1 : position des planètes
Astro 2 : distances terre soleil

de M. Pierre MALLER
Graph — Dessin de fonction

de M. BOISRAYON
PERAXO : le programme donne les coordonnées cartésiennes planes d'une projection axonométrique d'un objet spatial défini par les coordonnées tridimensionnelles cartésiennes d'un certain nombre de ses points.
PERCEA : Programme donne les coordonnées cartésiennes planes d'une projection perspective centrale ou cavalier d'un objet spatial défini par les coordonnées cartésiennes d'un certain nombre de ses points.

Titre L'ILE AU TRESOR numéro ILE EJ-1-B page 1 de 3 SHARP

Programmeur Didier BALMETTE date 18-04-82 cassette n° PC-1211

Description du programme

La mer est représentée par une grille 10x10 dont les rangées et les colonnes sont repérées par un chiffres de 0 à 9. Au début de chaque partie la machime cache a l'intérieur de la grille une ile dont elle vous donnera les coordonnées. De plus elle installe en un point un navirè qui vous appartient. Le but du jeu est d'aller sur l'ile appelé l'ile des morts a l'aide de votre navire pour y prendre le trésort des morts.

Pour cela a chaque tour la machine affichera le numero des vents qui sont disponible dans la région où se situe votre navire. Suivant la difficulté choisi vous aurait de 0 à 10 vent. Bien sur pour un niveau donné il pourra y avoir q'un nombre réduit de vents.

A l'aide d'un de ces vents vous vous déplacer d'une case dans le sens du vent et d'une seule.

Lorsque vous aurait atteind l'ile vous aurait gagné.

<u>Exemple</u>

	ile en 1.4	(abscisse, ordonnée)
difficulte	1	
vents	3 2 1 4	(sud.E-sud-sud.O-Ouest)
position bateau	7.0	
vent pris	4	
nouvelle pos	6.0	
ile	1.4	
bateau	6.0	
vents	2 8 3 6 7	(sud-nord-sud.E-est-nord.O)
vent	8	
position	6.1	
etc...		

mode d'emploi

n	procédure	introd.	press	affichage
1	Introduire le programme ou lire la cassette			
2	Lancer le programme	RUN	ENTER	L ILE AU TRE LES VENTS INITIAL
3	Introduire le générateur	Z	ENTER	DIFFICULTE
4	Introduire le degrée de dificulté	o ou 3	ENTER ENTER ENTER ENTER	POSITION DE POSITION DU VENTS DISPO NOUVELLE PO
5	Introduire la nouvelle position	X,Y	ENTER	*UNE SEULE 5 *GAGNE (6) *retour en 4
6				UNE AUTRE
7	Introduire O pour rejouer	O ou N	ENTER	*retour en3 *THE END

Titre L'ILE AU TRESOR date 18-04-82 page 2 de 3

Memoire :

n°	contenue
A	vents
B	
C	
D	
E	
F	
G	
H	
L$	
M	auxilière
N	auxilière
O$	"O ou N "
P	mL coord.
Q	boucle
R	boucle
S	nb vents
V	co. bateau
W	co. ILE
X	O ou 3
Y	O < Y < 1
Z	initial

ref. cassette

de à

Programme :

adr	touches	commetaires
1Ø	BEEP 1:PAUSE "+++ L^ILE AU TRESOR +++	
2Ø	PAUSE "LES VENTS SONT DESIGNES":PAUSE "PAR LES CHIFFRES :	
3Ø	PAUSE "NORD=8 N.O=7 N.E=9":PAUSE "SUD=2 S.O=1, S.E=3":PAUSE "OBEST=4 EST=6	
4Ø	INPUT "INITIAL ";Z:L$=" "	
5Ø	"O"BEEP 1:INPUT "DIFFICULTE (Ø a 3);";X	
6Ø	GOSUB 35Ø:W=Y	
7Ø	GOSUB 35Ø:W=W+.1Y	
8Ø	GOSUB 35Ø:V=Y	
9Ø	GOSUB 35Ø:V=V+.1Y	
1ØØ	M=INT V:N=INT W	
11Ø	IF (M=N)+(M-1=N-1)+(M+1=N+1) GOTO 6Ø	
12Ø	M=V-INT V:N=W-INT W	
13Ø	IF (M=N)+(M+.1=N+.1)+(M-.1=N-.1) GOTO 6Ø	
14Ø	FOR R=1 TO 8 :A(R)=O:NEXT R	
15Ø	GOSUB 36Ø:S=1+INT((8-2X)Z)	
16Ø	FOR R=1 TO S	
17Ø	GOSUB 36Ø:Y=1+INT(9Z)	
18Ø	IF Y=5 GOTO 17Ø	
19Ø	A(R)=Y:IF R=1 GOTO 225	
2ØØ	Q=1	
21Ø	IF A(R)=A(R-1) GOTO 17Ø	
22Ø	IF Q<>R-1 LET Q=Q+1:GOTO 21Ø	
225	NEXT R	
23Ø	BEEP 1:PRINT "POSITION DE L^ILE:";W	
24Ø	BEEP 1:PRINT "POSITION DU BATEAU/";V	
25Ø	BEEP 1:PAUSE "VENTS DISPONIBLES":PRINT L$;A;L$;B;L$;C;L$;D;L$;E;L$;F;L$;G;L$;H	
26Ø	INPUT "NOUVELLE POSITION ? ";P	
27Ø	M=INT P:N=INT V	
28Ø	IF (M>M+1)+(M<N-1) BEEP 1:PAUSE "UNE SEULE CASE":GOTO 26Ø	
29Ø	M=P-INT P:N=V-INT V	

partition MEMOIRES pas

Titre L'ILE AU TRESOR   date 18-04-82   page 3 de 3

Memoire :   Programme :

n°	contenue

adr	touches	commetaires
300	IF (M>m+.1)+(M<N-.1) BEEP 1:PAUSE "UNE CASE SEULEMENT":GOTO 260	
310	IF P<>W 1ET V=P:GOTO 140	
320	BEEP 2:PAUSE "GAGNE,L'ILE EST A VOUS ":INPUT "UNE AUTRE PARTIE (O/N)?"; O$:GOTO O$	
330	"N"BEEP 1:PAUSE " THE END":END	
350	GOSUB 360:Y=INT(10Z):RETURN	
360	Z=Z+ :Z=ZZZZZ:Z=Z-INT Z:RETURN	
370	END	

TIHP « Le château de tihp »

Pour écrire ce programme, je me suis inspiré d'un article paru dans Jeux et Stratégies (pgm écrit pour une TI57).

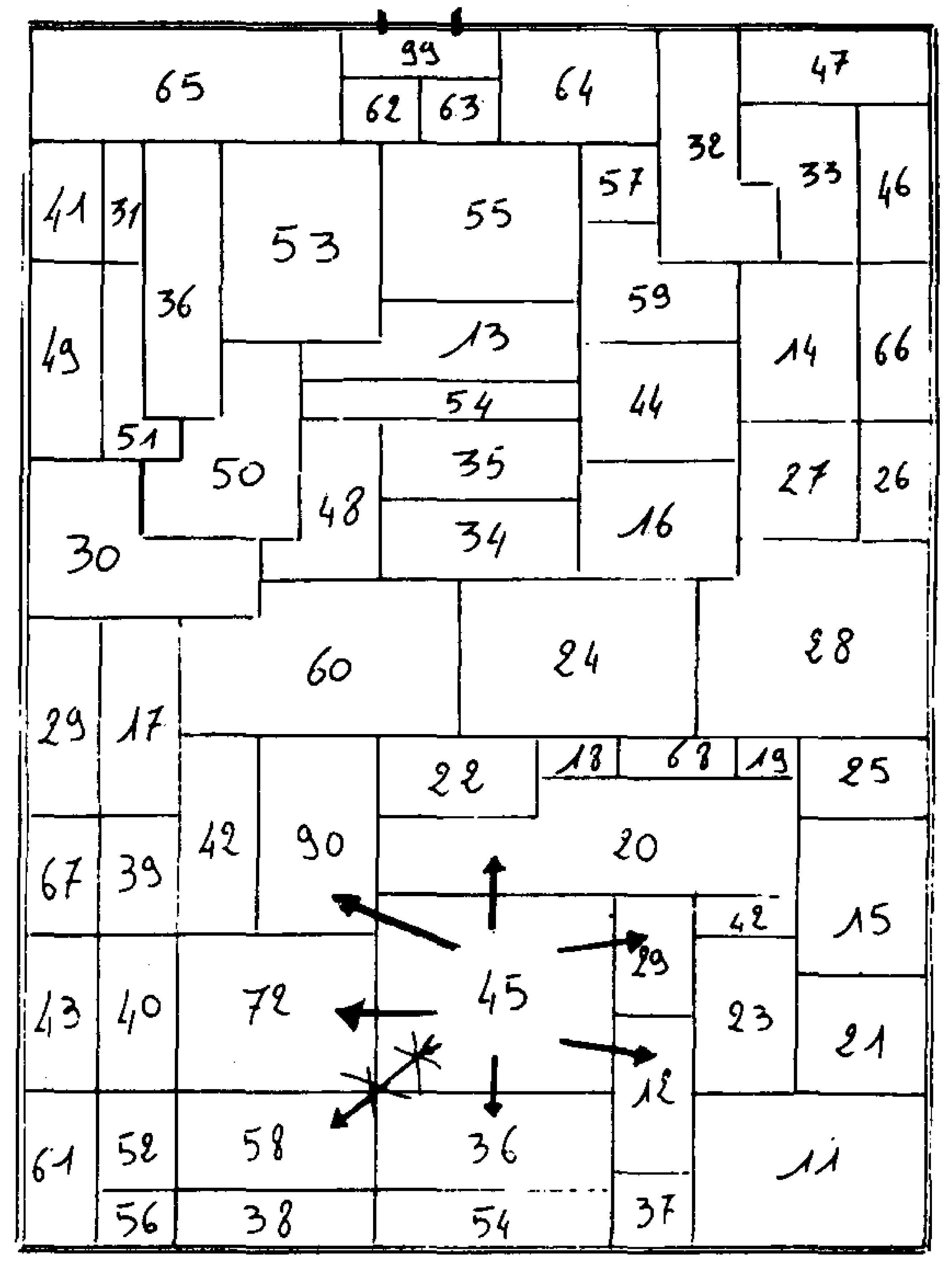
PGM.

```
  5:REM "TIHP"
 10:"=":CLEAR :D
    =3:E=5:B=11:
    Y=B
 15:A=(A+π)^5:A=
    A-INT A:G=
    INT (3A+1):X
    =E:E=D:D=X:Z
    =20:GOTO 45
 20:X=G:Z=25:
    GOTO 45
 25:B=F
 30:BEEP 1:IF Y=
    BPAUSE " LA
    PORTE S OUVR
    E !!!":GOTO
    40
 35:PAUSE " LA P
    ORTE RESISTE
     !!!"
 40:PRINT "POSIT
    ION : PIECE
    N. ";B:Z=B:
    GOTO 55
 45:X=1/X*F:X=X-
    INT X:IF X=0
    THEN 30
 50:GOTO Z
 55:INPUT "DESTI
    NATION : PIE
    CE N. ";F:Y=
    F:GOTO 15
```

C'est le mystérieux château de Tihp d'où il n'est pas facile de s'échapper ! Prisonnier vous tentez toutefois pour la Xième fois de vous évader. Ce n'est pas si simple. Certaines portes ont leur « sésame ouvre-toi », qu'il vous faudra trouver... Mais, certaines autres ne s'ouvrent jamais ! Etes-vous prêt à tenter l'aventure ?

LE JEU : Il commence par la réalisation du plan d'un château. Il pourra comporter un nombre quelconque de pièces, que vous numéroterez une à une. Tous les nombres entiers positifs peuvent être utilisés. La case de départ sera notée 11. Vous devrez considérer que toutes les pièces adjacentes sont reliées par des portes. Vous serez amené à rencontrer des portes qui s'ouvrent toujours, d'autres jamais et d'autres encore qui s'ouvrent... parfois. Les dernières peuvent d'ailleurs fort bien se refermer juste après votre passage et ne plus s'ouvrir. Concrètement pour tenter de passer d'une pièce à une autre, en direction de la sortie, le joueur devra introduire le nombre inscrit dans la pièce dans laquelle il désire aller. Pour autant que celle-ci est adjacente à celle où il se trouve. Si le joueur est coincé dans une pièce ou dans une partie du château, il pourra tenter de « forcer » des portes. C'est-à-dire une nouvelle tentative pour aller par une porte qui s'était révélée close. Après plusieurs parties vous parviendrez à saisir la logique de l'ouverture et de la fermeture des portes.

MEM : 1148 STEPS ; 143 MEMORIES.



TOUCHES	AFFICHAGE	NOTE
	LA PORTE S'OUVRE !!! (pause) POSITION : PIECE N° 11	Position de départ
SHFT =	DESTINATION : PIECE N°	
ENTER 23 (Enter)	LA PORTE S'OUVRE !!! (pause) POSITION : PIECE N° 23	
	DESTINATION : PIECE N°	
ENTER 42 (Enter)	LA PORTE RESISTE !!! (pause) POSITION : PIECE N° 23	
ENTER 29 (Enter)	DESTINATION : PIECE N° LA PORTE S'OUVRE !!! (pause) POSITION : PIECE N° 29	La case d'arrivée est la 99ème pièce, la seule qui débouche sur le parc du château. THE END.

Note : *"Faites attention aux oubliettes"*.

Nouveaux Produits PC-1251

– Une nouveauté.

– Avec le 1500 nous avions été étonnés de la prouesse technique des Ingénieurs SHARP. Réussir à mettre la puissance d'un ordinateur de table dans un ordinateur de poche, c'était déjà un bel exploit. Mais avec le PC-1251 on se rend compte que l'impossible n'a pas de limite car la taille de cette machine est impressionnante. A quand l'ordinateur de poche dans une montre ?

– Esthétiquement la 1251 surclasse ses grandes sœurs. Fini les poches déformées. Présenté dans un petit carnet elle est légère et discrète. (135 x 70 9,5 mm).

– Le clavier est formé en 3 parties : un pavé numérique avec des touches larges et le clavier QWERTY, qui possède des toutes petites touches mais dont l'éloignement permet désormais une frappe aussi rapide que sur la 1211.

– L'afficheur est composé de 24 matrices 5 x 7 et de 8 indicateurs, de plus pour permettre une lecture sous tous les angles l'intensité de ce dernier est réglable. Innovation le changement de mode s'obtient par un bouton poussoir.

– Passons maintenant à la machine elle-même ; celle-ci est composée d'une MEM de 24 KO et d'une MEM de 4348 octets répartis ainsi :

— 308 octets pour les mémoires fixes (A,Z,A,$,Z)
— 48 octets pour la mémoire réserve
— 3486 octets pour la zone programme, donnée.
— 606 octets pour la zone système.

– Ce basic de 1251 allie les avantages du 1211 et la puissance du 1500. On peut dire que ce basic est ni plus ni moins celui du 1500. Mais sans le graphique, « horloge et le rafraîchissement de l'affichage », pour tout ce basic fait 24 KO et devrait donc être plus imposant. Il l'est car il possède beaucoup d'instructions inutilisables pour le moment. Ainsi on trouve des instructions de RS-232 C tels que COM $, SETCOM, INSTAT, OUTSTAT, ERROR, on trouve aussi des instructions ouvrant l'accès au langage machine (qui d'ailleurs est différent de celui du PC-1500) PEEK, POKE, CALL et d'autres instructions tels que OFF KEY ROM C... et O...
Il s'agit donc d'un basic relativement complet pour un micro poche ; vous trouverez les commandes et instructions plus loin.

– Cette machine déjà performante voit ses possibilités accrues par une interface imprimante micro-cassette d'une taille si réduite que la portabilité du système est conservée (205 x 149 23 mm). L'imprimante est thermique et imprime 24 caractères par ligne à raison de 0,8 lignes par seconde. Situé en regard de l'afficheur on contrôle directement les sorties de données sur celle-ci.

– L'interface microcassette sera bien pratique ; fini les problèmes d'incompatibilité de magnétophone. Celui-ci est incorporé et utilisé des **microcassettes TM modèle courant en Europe.**

– Notez que la 1251 accepte directement soit si vous utilisez des microcassettes en plaçant votre cassette dans son magnéto soit en branchant votre magnéto sur la prise d'entrée tous vos programmes pour PC 1211.

– Appartenant à la famille SHARP, la 1251 se verra dotée à son tour d'un club d'utilisateurs.

PC-1211, PC-1500, PC-1251. A quand le PC-1540 ?

CE 159

Posséder 8 KoMEV dans un 1500 c'est pratique mais si l'on veut avoir plus de 8 Ko il faut acquérir l'interface K 7 qui en stockant les programmes sur bandes magnétiques étend la taille de votre machine. Maintenant une autre solution s'offre à vous, le CE 159.

Le CE 159 est une extension mémoire de 8 Ko (jusqu'ici rien de nouveau) mais on peut protéger sur ce dernier la mémoire par pages de 2 Ko.Contenant une pile CR 2032 le module conservera intact vos programmes même hors du PC 1500 sachant qu'il faut environ 10 secondes pour changer de modules, on atteint une vitesse de 6.553,6 bauds.

Dans la 1500 le module est alimenté par la machine elle-même et la durée de vie de la pile est alors de 5 ans. Si ce module est retiré du 1500, la durée de vie de la pile, donc de la conservation de vos programmes, est de 2 ans.

Alors voici un nouveau module mémoire qui rendra service à ceux qui veulent développer du logiciel ou bien économiser l'achat d'une interface cassette.

CE 153

La tablette de logiciel : il s'agit d'un second clavier utilisable uniquement par programme comportant 140 touches sensitives.

Pour être utilisé, il faut changer en mémoire un programme en langage machine de 1 KO. Après l'utilisation de ce nouveau clavier est très aisé. Mais à quoi peut-il bien servir ???

C'est à vous de choisir et de créer pour cette interface un logiciel performant mais les exemples d'utilisation sont nombreux.

– Jeux : clavier utilisé en échiquier ; déplacer les pièces, le 1500 se chargera du reste.

– Musique : il fera un clavier d'orgue idéal.

– Gestion : adressage direct d'un article.

– Scientifique : entrée de données dans un ...CALC.

– Enseignement : apprenez une langue utilisant un autre alphabet ou bien reliez des dessins ou des mots.

Tous à vos claviers et vous verrez qu'une tblette peut être très utile dans certaines applications.

CE 158

Du nouveau pour la PC-1500.

– Votre machine va pouvoir aggrandir ses connaissances. Désormais le monde extérieur lui est ouvert grâce au CE 158, la RS-232 C que SHARP a conçu.

– Le RS-232 est une interface série permettant la transmission de données entre le PC-1500 et un périphérique extérieur, standard EIA RS 232 C. Pouvant aussi bien se brancher sur le 1500 ou sur le CE 150, le système conserve son format de poche.

– Comme presque toutes les interfaces SHARP, le CE 158 apporte son BASIC, un basic puissant qui fait de cette interface un outil performant. Tout est programmable par logiciel.

La vitesse qui peut être de 50, 100, 110, 200, 300, 600, 1200 ou 2400 bauds.

La parité paire, impaire ou inexistante.

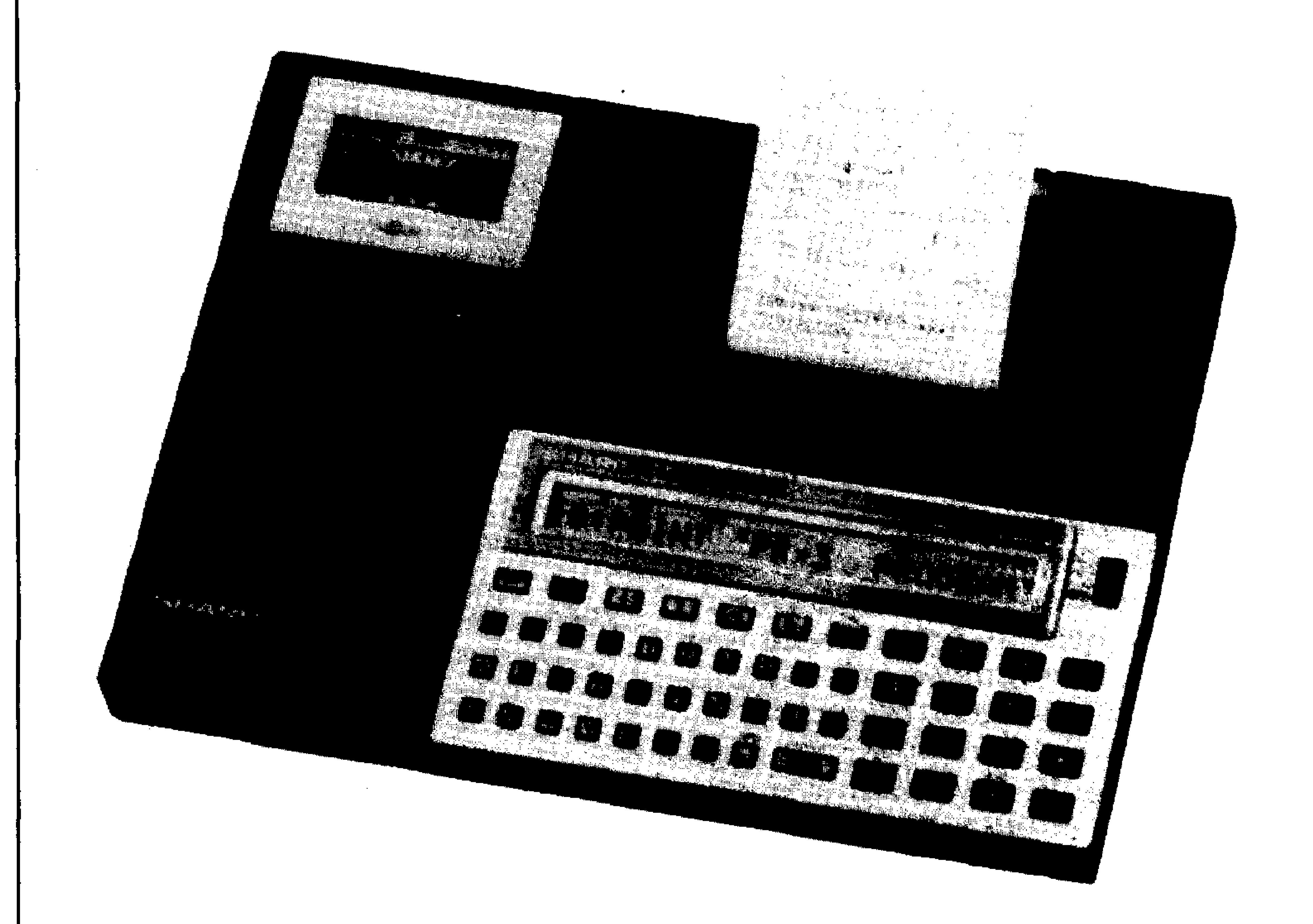
La longueur 5, 6, 7 ou 8 bits.

Le bit d'arrêt (1 ou 2).

En plus du basic le CE 158 possède dans ses mémoires un programme TERMINAL qui permet l'utilisation de votre 1500 comme un terminal. Sans programmation vous pouvez envoyer et recevoir ds données. Tout était déjà très bien, le série c'est intéressant, mais la parallèle aussi, alors SHARP a inclus en interface parallèle au CE 158 (prévu pour connecter l'imprimante SHARP-CE 332-P).

– Merveilleux petit engin que ce petit boitier (86 x 115 50 mm) qui, accompagné d'un manuel en quatre langues vous initiera aux joies de la communication. Notez aussi qu'il est fourni dans ce manuel un programme de transmission de données entre le MZ 80 B et le PC-1500 et que celui-ci permet de sauvegarder des programmes, le 1500 sur diskettes (par l'intermédiaire du MZ 80 B) et ce, 12 fois plus rapidement qu'en employant le CE 150.

Alors à vos modems, ordinateurs, tables traçantes ou appareils de mesures et élargissez le savoir de votre PC-1500.



EDITORIAL MZ
Le bulletin nouveau est arrivé...

350 adhérents, autant de bulletins envoyés, plus de 500 programmes et lettres expédiées ; est-ce suffisant pour excuser le "léger" retard dans la parution de notre bulletin numéro 2 ? C'est vrai, nous n'avions pas escompté un tel succès et avions prévu un peu juste quant à l'équipe et aux moyens mis en place. L'efferverscence du SICOB passée, nous avons obtenu, de la part de S.B.M., la possibilité de doubler, en effectif et en moyens matériels, le potentiel de l'équipe mise à la disposition du Club ; cela devrait nous permettre de resorber le retard actuel du courrier avant la fin novembre, et de pouvoir répondre dans un délai maximum de 15 jours à toute demande à partir de décembre.

Autre modification importante : le cadeau de bienvenue accordé à tout nouvel adhérent a dû être modifié ; cela pour 2 raisons : la recherche et l'enregistrement de 5 programmes à chaque nouvelle adhésion prenaient environ 45 mn et contribuaient a retarder d'autant les réponses aux autres demandes ; l'octroi de ces programmes a provoqué une épidémie de multiadhésions pour un même MZ(jusqu'à 11 adhésions pour le même appareil !) qui vont à l'encontre de l'esprit qui anime le Club. Ces 2 motifs nous ont conduits a réduire à 1 programme choisi dans la logithèque le cadeau de bienvenue accordé à tout nouvel adhérent inscrit depuis le 1.10.82. Ce cadeau reste, par contre, renouvelable tous les ans, à la date d'inscription de l'adhérent.

***

QUELQUES PRECISIONS SUR LE FONCTIONNEMENT DU CLUB.

** Toute demande de programme doit être accompagnée d'une cassatte qui ne doit surtout pas être de haute qualité, il est en effet fort probable que vous ne receviez pas la même cassette en retour.*
** Pour tout programme reçu, l'octroi de points reste le même que celui indiqué dans le bulletin n° 1, soit : note donnée au programme multipliée par 2.5.*
** Il est souhaitable que chaque programme envoyé soit accompagné d'une notice d'utilisation qui sera jointe à la cassette contenant ce programme ou directement incorporée au programme.*
** Eviter, dans la mesure du possible, de n'envoyer que la liste du programme ; le recopier prend du temps et provoqué inévitablement des erreurs qui ne sont pas toujouts évidentes à trouver dans un programme que l'on ne connaît pas.*
** La permanence du Club se tiendra dorénavent et définitivement tous les mercredis, de 14 à 18 heures ; les visites ou les coups de téléphone y sont les bienvenus.*
** Tout adhérent peut nous faire parvenir un article d'intérêt général ou traitant d'un sujet particulier étudié ou mis au point par son auteur. Cet article sera édité dans le bulletin suivant et sera (largement...) rémunéré en points-programmes.*

A la demande de beaucoup d'entre vous, nous mettons actuellement au point le projet d'une réunion nationale regroupant en un même lieu pour une journée, tous les membres du Club MZ. Cette réunion n'aurait d'intérêt que si une grande majorité de MZistes de province acceptaient de faire le déplacement dans cet unique but. Il nous faudrait donc, pour ce projet, tous vos avis. MZistes de province, veuillez donc nous faire connaître votre opinion, avant la fin de l'année, sur le principe d'une éventuelle rencontre de tous les adhérents de France qui se déroulerait vers le mois de mars 1983 à Paris ou dans sa proche banlieue. Si ce projet recueille suffisamment de suffrages, la date et le lieu de cette rencontre vous seront communiqués dans le prochain bulletin.

Vous le constatez, après une inévitable période de rodage, le Club commence à prendre sa vitesse de croisière. Tout n'est pas encore parfait mais nous faisons tout ce qu'il faut, ne serait-ce que dans la présentation de ce bulletin n° 2 pour rendre aussi parfaite que possible la communication entre MZistes.

Permettez moi de souhaiter dès maintenant de très bonnes fêtes de fin d'année ainsi qu'une excellente année 1983 à votre vieil ami MZ ainsi qu'à vous-même.

Sylvain Bizoire

Pour les étourdis...

PRINCIPES ET FONCTIONNEMENT DU CLUB : Le Club est ouvert à tout possesseur de MZ 80 K, B, A qui nous fait parvenir la demande d'inscription située en dernière page, remplile et signée.

L'adhésion au Club est totalement gratuite et ne donne lieu a aucune obligation ni engagement d'aucune sorte.

Chaque adhérent recevra, 4 fois par an, le bulletin du Club où seront recensées toutes les astuces, suggestions, idées, ainsi que les programmes émis par l'ensemble des adhérents.

Pour tout problème particulier, il est également possible à chacun de questionnner votre serviteur, soit par courrier, soit par téléphone (le mercredi de 14 à 18 h) ; il fera le maximum et mettra, éventuellement, l'équipe SHARP a contribution pour tenter de vous donner satisfaction.

ET LE PROGRAMME ? La procédure concernant les échanges de programmes, adoptée après hésitations, répond à 3 objectifs :

1 Posséder des logiciels de qualité en accordant une légitime contrepartie aux créateurs de ces logiciels.
2 Faire profiter de ces programmes, par une répartition équitable, l'ensemble des adhérents, permettant ainsi aux MZistes débutantsde bénéficier de l'acquis des programmeurs chevronnés et de le mettre en pratique dans leurs applications futures.
3 De créer, par le biais de ces échanges, un "ESPRIT CLUB", amical et solidaire, trait d'union entre tous ses membres.

Dans cette optique, nous vous proposons les modalités suivantes pour nous envoyer ou vous procurer des programmes :

Chaque programme reçu d'un adhérent se verra attribuer un nombre d'étoiles (de 1 à 10) ; en contrepartie, le concepteur de ce logiciel bénéficiera d'un bonus égal à 2,5 fois le nombre d'étoiles attribuées, qu'il pourra utiliser pour puiser dans la logithèque.

En guise de cadeau de bienvenue, chaque adhérent aura à son crédit, dès son inscription, 10 étoiles qui lui permettront d'accéder à la logithèque ; les éventuels bonus gagnés par l'envoi de programmes personnels venant, bien sur, s'ajouter à ce crédit. Ce cadeau sera renouvelé tous les ans, à la date d'inscription de l'adhérent.

Pour commander un ou plusieurs programmes de la logithèque. Il suffit d'avoir à son crédit un nombre d'étoiles au moins égal au total des étoiles demandées pour chaque programme commande.

MZ 80 K

Super basic... suite...

Bernard KOKANOSKY, seul utilisateur du SUPER BASIC à ne pas trouver son BASIC assez super lui a ajouté "quelques" modifications et instructions nouvelles et l'a logiquement intitulé SUPER BASIC 2. En plus des caractéristiques décrites dans le bulletin de juin, la version 2 possède :

– Les ordres SET, RESET, POINT, MOVE, PLOT, HRES qui permettent de programmer directement en BASIC la carte

haute résolution "QUANTUM MICROS" avec une résolution de 200 * 320 points. De plus, on obtient, avec HRES les caractères "e", "i" et "a" accentués ainsi que le "c" cédille.
- Possibilité d'inversion vidéo.
- Nombre de boucles FOR...NEXT imbriquées illimité ; cela permet de les utiliser en recursivite quelle que soit la profondeur de la recursivité.
- etc.

Vous étonnerai-je en vous apprenant qu'une troisième version de SUPER BASIC est à l'étude ? Il s'agit d'une version DCB ("Decimal Code Binaire" diraient les puristes) ; elle permet une plus grande précision dans les calculs mathématiques :

- Calcul sur 13 chiffres,
- affichage sur 10 chiffres,
- exposants entre + 99 et - 99,
- fonctions transcendantes sur 15 chiffres calculées par le C.O.R.D.I.C.
- et j'en passe...

Cette version n'est encore qu'une ébauche ; elle ne verra le jour que si elle intéresse un nombre suffisant d'utilisateurs, le travail demandé pour son élaboration étant considérable. Matheuses, Matheux, à vos plumes, la naissance de SUPER BASIC DCB ne dépend que de vous.

Pourquoi ne pas s'attaquer, dans la foulée, à un BASIC existant, le 5025 ?, sitôt dit, sitôt modifié ; le 5025 KOKANOSKI était crée et améliorait sensiblement les possibilités du SP 5025 original :

- GOTO et GOSUB peuvent être suivis par une expression mathématique (EX : GOSUB 2*B+12), sauf pour ON..GOTO et ON..GOSUB,
- Idem pour RUN et THEN mais l'expression doit commencer par un chiffre,
- RND est modifié de façon à donner des résultats un peu plus aléatoires,
- L'erreur PRINT 1E+00 donnant 10 est corrigée,
- L'interférence entre TI$ et LIMIT est supprimée,
- PEEK est débloqué,
- GET et INPUT sont corrigés : pluls d'erreur ou de DATA ERROR quand on app tape "+" ou "-" à la suite d'un GET ou d'un INPUT,
- RESTORE peut être suivi d'un numéro de ligne ou d'une expression mathématique si le premier caractère est un chiffre.

Comme tout bon langage se doit de posséder ses programmes d'application, son auteur nous a fait parvenir une série de plusieurs exemples d'utilisation, essentiellement mathématiques, qui vous permettront d'acquérir plus rapidement le style de programmation SUPER BASIC.

* Interpolation polynomiale,
* Calcul sur les séries,
* Graphes y = f(x),
* Courbes paramètrées,
* Courbes de lissajous,
* Racines de f(x) = 0,
* Racines de P(x) = 0 (P, polynome . Racines dans C),
* Etude sur la résolution des équations,
* Convergence en loi de B(n,p) et P(1) vers N(0,1),
* Illustration de la loi des grands nombres (Prob.),
* Illustration des fluctuations aléatoires (Prob.),
* Calcul de -ln[1-(a-b)] (Precision multiple),
* Calcul des factorielles (exact mais long !) (Précision multiple),
* Calcul de πe (Précision multiple),
* Tri selon méthode de Shell,
* Systèmes d'équations non linéaires,
* Equations différentielles (Méthode de RUNGE KUTTA RK 44),
* Algèbre linéaire,
* Haute résolution en 3 dimensions (Carte QUANTUM MICROS),
* Graphes y = f(x) (Carte QUANTUM MICROS).

Chacun de ces programmes coûte 3 points ; certains d'entre eux sont d'ailleurs listés dans la notice qui accompagne chaque cassette SUPER BASIC.

2 remarques pour terminer :

- Il existe une version "éditée" de SUPER BASIC, c'est-à-dire facilement lisible et transformable. Avis aux possesseurs de MZ 80 A, habitués de l'assembleur, que l'adaptation sur leur appareil tenterait...
- Est-il utile de rappeler que la conception de tels programme nécessite une parfaite connaissance du microprocesseur Z 80 et de son langage, ainsi que de la structure interne du MZ 80 K. Bernard KOKANOSKI mérite, à ce titre toute notre admiration et notre respect.

Editeur assembleur.

A la suite de nombreuses demandes, j'ai réalisé une traduction résumée de la notice des 3 programmes EDITOR ASSEMBLER & TEXT EDITOR SP qui permettent de programmer directement, dans son propre langage, le microprocesseur Z 80 qui équipe tous les appareils de la série MZ.

Cette notice est également compatible pour les 3 programmes travaillant sous F DOS pour le MZ 80 B ; la différence éssentielle étant que ces derniers enregistrement sur disquette et non sur K7.

Toutes critiques ou suggestions concernant la poursuite éventuelle d'une série d'articles sur le langage assembleur des différents MZ sont les bienvenues.

ASSEMBLEUR

(1) UTILISATION DE L'EDITEUR ASSEMBLEUR
1) Editer le programme ou sous-programme en tapant, en clair, les instructions en langage machine du Z 80 et en respectant les règles de l'assembleur (voir "COMMANDES DE L'EDITEUR").
2) Assembler le programme composé par "PASS 1" et corriger les éventuelles erreurs d'assemblage (voir MESSAGES D'ERREUR) en retournant à l'éditeur (PASS X).
3) Enregistrer le *Programme assemblé* pour liaisons ultérieure éventuelle avec d'autre sous-programmes et transformation en *Programme machine*.

(2) COMMANDES DE L'ASSEMBLEUR
PASS 1 : Assemblage du *programme source* créé avec l'éditeur de texte.
PASS 2 : Liste du programme assemblé sur écran. Peut être interrompue par [SPACE].
PASS 3 : Liste du *programme assemblé* sur imprimante.
PASS 4 : Enregistrement du *programme assemblé*.
PASS X : Retour à l'éditeur de texte.
PASS ! : Retour au moniteur.
GOTO $ 2200 = Retour à l'éditeur.
GOTO $ 2260 = Retour à l'assembleur.

(3) DEFINITIONS DU LANGAGE ASSEMBLEUR
LABEL : Titre de 6 lettre maxi qui définit un sous-programme, une adresse ou une donnée. Doit être placé à chaque fois que l'on a besoin d'une référence.
LABEL : définit un sous-programme.
LABEL :EQU définit une adresse ou une donnée.
LABEL : ENT définit un sous-programme qui peut comprendre plusieurs autres sous-programmes ou LABELS. Doit être déclaré pour chaque sous-programme édité afin de faciliter les liaisons par K7 avec d'autres sous-programmes.
REL : Fait démarrer le programme à l'adresse indiquée (à éviter, provoque des complications lors des transferts de programmes sur K7).
DEFBn : Donne une valeur à l'adresse ou est indiquée l'instruction ; en Hexa (H), Decimal (D), ou code ASCII (" ").n<255(d),n<FF(H). 1 octet adresse.
DEFWnn' : Idem a DEFB, mais sur 2 octets. Ex. : 1200 DEFW 01FOH ; 1200 = FOH, 1201 = 01H. n≤65535(D), n≤FFFF(H).
DEF"M" : Définit une "CHAINE" de 64 caractères ASCII maxi ; place chaque lettre de la "CHAINE" dans les adresses qui suivent l'instruction.
DEFSnn' : Réserve une place mémoire égale a nn' octets pour y placer des données.
SKPn : Laisse un blanc de n lignes dans la liste d'assemblage.
SKPH : Laisse une page blanche.
END : Termine obligatoirement tout *programme source* destiné à être assemblé.

(4) MESSAGES D'ERREUR DE L'ASSEMBLEUR
E — Indique que le LABEL nommé sur la ligne n'est pas défini dans le programme. (Peut l'être dans un autre sous-programme),
P — Indique qu'un LABEL:EQU doit être placé en début de programme pour pouvoir être lié, ensuite, à d'autres sous-programmes,
C — Caractère illégal ou erreur de syntaxe dans l'opérande,
F — Erreur de format dans l'instruction,
N — Indique que ENT ou EQU ne sont pas accompagnés de leur valeur ou symbole,
L — Le symbole utilisé dans l'instruction n'existe pas dans le programme en cours. (Peut exister dans un autre),
M — Indique qu'un LABEL est utilisé plus d'une fois,
O — Indique une erreur dans l'opérande ou l'instruction,
Q — Indique une erreur d'ortographe ou de syntaxe dans l'instruction,
S — Indique qu'une chaîne de caractères n'est pas entourée de " " ou ' ',
V — Indique que la valeur de l'opérande dépasse les possibilités de l'instruction.

EDITEUR DE TEXTE

(1) UTILISATION DE L'EDITEUR DE TEXTE

L'EDITEUR DE TEXTE
- Met en mémoire tous les caractères ("CR" compris) tapés par le programmeur, à la suite les uns des autres,
- Enregistre ou lit un *programme source*
- Permet des corrections à partir de CP (compteur de programme) qui est toujours placé ENTRE deux caractères.

LE PROGRAMME SOURCE enregistré sous éditeur n'est UNIQUEMENT utilisable que pour un assemblage ultérieur par l'assembleur.

UNE LIGNE DE PROGRAMME SOURCE comprend obligatoirement tout ou partie des éléments suivants :
LABEL: CODE OPERATION [SPACE] ADRESSE OU DONEE [SPACE] ; COMMENTAIRE
ex : MONIT: JP OOOOH ; Saut moniteur.
Bien respecter la ponctuation (:;) et les [ESPACES].

COMMANDES DE L'EDITEUR DE TEXTE

LECTURE
R - (+ TITRE) Lit un programme source, le place en début de mémoire et place CP en tête du programme.
A - (+ TITRE) Lit un programme source, le place après CP préalablement placé à la fin du programme précédent. Permet de charger et de lier plusieurs programme à la suite.

AFFICHAGE
T - Affiche le contenu de l'éditeur sur écran.
nT - Affiche n lignes à partir de la position de CP.

MOUVEMENTS DE CP
B - CP est placé au début de la mémoire éditeur.
Z - CP est placé à la fin du programme en mémoire.
nJ - CP est placé en tête de la ligne numéroté n.
nL - CP est placé en tête de la n'ème ligne qui suit celle qu'il occupe.
L - CP est placé en tête de la ligne qu'il occupe.
nM - Déplace CP de n caractères ([space] ou [CR]) qui suivent la position qu'il occupe.

MODIFICATIONS
C - C CHAINE$1→CHAINE$2 recherche la chaîne de caractères CHAINE$1 depuis la position de CP vers la fin du programme ; quand il l'a trouvée, la remplace par CHAINE$2 ; CP est placé après la nouvelle chaîne.

Q - Q CHAINE$1→CHAINE$2 idem a C mais sur toutes les CHAINE$1 qui suivent CP, jusqu'à la fin du programme. CP est placé derrière la dernière CHAINE$1 remplacée.

C CHAINE$1 ou Q CHAINE$1 efface toutes les CHAINE$1 au lieu de les remplacer.

I - I CHAINE$1→[CR] place CHAINE$1 après CP. Si pas de [→] : Nouvelle lign ; CP est placé après CHAINE$1. Si [→] avant [CR] : Fin de l'instruction I.

UTILISER I POUR L'ENTREE DU TEXTE DANS L'EDITEUR

K - Efface tous les caractères situés après CP jusqu'à [CR] qu'il n'efface pas. (fin de ligne)
nK - Efface tous les caractères situés après CP (nK) ou avant CP (-nK), sur n lignes.
nD - Efface n caractères après CP (nD) ou avant CP (-ND), [CR] compris.
S - S CHAINE$1 recherche CHAINE$1 à partir de la position de CP puis CP se place juste derrière CHAINE$1.

ENREGISTREMENT
W - (+ TITRE) écriture d'un *programme source* sur K7.
V - Vérifie l'enregistrement.

SPECIAL
= -Donne le total des caractères ([SP]&[CR] compris) contenus dans l'éditeur.
. -Donne le numéro de ligne ou se trouve CP.
& -Met à 0 toute la mémoire éditeur.
X -Passe à l'assembleur.
! -Retour au moniteur.
GOTO$2200 : Mémoire éditeur vide,
GOTO$2260 : Programme conservé.

EDITEUR DE LIENS (RELOCATABLE LOADER)

(1) UTILISATION DE L'EDITEUR DE LIENS
L'éditeur de liens permet de lier plusieurs sous *programmes assemblés*, de créer une table de symboles commune à tous les sous-programmes et d'enregistrer sous forme définitive un programme en langage machine directement éxécutable à partir du moniteur ou d'un ordre BASIC USR. L'éditeur de liens ne permet aucune correction ; il enregistre, avec la commande "S", un * programme édité*.

(2) COMMANDES DE L'EDITEUR DE LIENS
TITRE
TBL - Adresse du départ de stockage de la table des LABELS. Est demandé dès le chargement de l'éditeur de liens.
ex : ** RELOCATABLE LOADER SP-2301* * (Titre de l'éditeur)
LINK AREA 1E4C - CFFF (Mémoire disponible)
***TBL BO (La mémoire disponible pour la table des LABELS va de BOOOH à CFFFH soit = 8 Koct.)**

CHARGEMENT
L - Lecture du programme assemblé + adresse de chargement dans la surface libre indiquée par "LINK AREA" ; le tout en Hexa.
ex : *LL 30000 2000 + titre éventuel.
N - Lecture et liaison d'un second (ou n'ème) *programme assemblé* avec le (les) programme précédent.

COMMANDES
H - Affiche les adresses Hexa de départ et de chargement d'un programme qui succèderait à ceux déjà entrés. (Utile pour "TO?" de "S".)
T - Affiche le contenu de la table des LABELS avec les messages correspondants :
U = Label indéfini (adresse ou donnée)
M = Label défini 2 ou plusieurs fois. (adresse ou donnée)
X = Labels qui se recoupent ou ont la même signification.
H = Label incomplétement défini (Donnée)
D = Label qui indique une donnée. N'est pas une erreur.
* -Remet à zéro les valeurs de départ et de chargement des programmes, ainsi que le contenu de la table des symboles ; sauf l'adresse TBL.

ENREGISTREMENT
S - Enregistrement du programme définitif (*Programme édité*)
FROM? : Adresse de chargement du 1er. * Programme assemblé*
TO? : Adresse de fin du dernier *Programme assemblé* (Visible par "H")
EXECUTE? : Adresse d'éxécution du futur *Programme édité*.
DATA? : Adresse à partir de laquelle sera chargé le futur *Programme édité*.
FILENAME? : Titre éventuel.
"FROM?" et *"TO?" :* concernent l'implantation ACTUELLE du programme en mémoire.
"EXECUTE?" et *DATA? :* concernent l'implantation et l'éxécution du futur programme machine.
V - Vérifie l'enregistrement du programme.
! - Retour au moniteur
GOTO$1200 Efface la mémoire programme.
GOTO$1260 Conserve le programme.

CORRECTEUR SYMBOLIQUE (SYMBOLIC DEBUGGER)

(1) UTILISATION DU CORRECTEUR SYMBOLIQUE
Le correcteur permet la correction des erreurs du programme lu en plaçant des points d'arrêt (P.A.) qui stoppent l'éxécution du programme et placent le contenu de tous les registres du Z 80 dans une zone mémoire qui peut être lue et corrigée. Le programme peut ensuite redémarrer avec les corrections effectuées.

CHARGEMENT
L - Charge le *Programme assemblé* + adresse de démarrage du programme + adresse de chargement (ex : *DL 3500 3000). Les 2 adresses (Hexa) doivent être supérieures à 2COOH + titre éventuel.
N - Charge le programme suivant qui succède et qui est lié au(x) programme(s) précécent(s) + titre éventuel.

FONCTIONS
H - Idem à "H" (Editeur de liens)
T - Idem à "T" (Editeur de liens)
* - Idem à "*" (Editeur de liens)
B - Mise en place de 1 ou plusieurs points d'arrêt (9 maxi)
1) Afficher le P.A. par son adresse Hexa ou un LABEL ou un LABEL (+)un déplacement décimal.
2) Espace.
3) Afficher le compteur de boucles si le P.A. est placé dans une boucle.
N = 1 si pas de boucle.
N > 1 et < 14 si le P.A. est placé dans une boucle ; le programme s'arrêtera à la boucle N.
Pour effacer un P.A., inscrire a nouveau son adresse suivie de "0" dans le compteur de boucles.

& - Efface tous les points d'arrêt.
M - Affiche le contenu d'une zone d'adresses mémoire, en Hexa, par lignes de 8 adresses. (ex : *DM 3300 - 3500), (ex : * DM LABEL1 - LABEL2) ; idem du LABEL1 au LABEL2.
Il est possible de corriger un octet de mémoire, comme en BASIC, en utilisant les déplacements du curseur. Toute correction doit être suivie de [CR].
Pour sortir de l'instruction, faire [CR] sur une ligne vide.
D - Affiche une partie de la mémoire programme en Hexa ; chaque ligne indiquant le nombre d'octets que comprend l'instruction lue. Définition des limites idem à "M".
[SPACE] interrompt la liste. [SCHIFT] [BREAK] passe à une autre commande.
W - Permet d'entrer en mémoire des instructions en Hexa ou par LABEL. En cas d'erreur, taper [SHIFT] [←], l'adresse s'affiche de nouveau pour correction.
G - Exécution du programme.
* DG 3200 éxécute le programme à partir de l'adresse 3200H.
* DG LABEL idem à partir du LABEL mentionné.
* DG [CR] idem à partir du P.A. avec le contenu des registres corrigés par l'instruction "B".
I - Indique la valeur de tous les registres du Z 80 et demande si c'est OK? ; si OUI → [CR] ; si NON → [SHIFT] [BREAK] et correction avec les instructions A, C ou P.
A - Affiche les registres A, F.
A - Affiche les registres A, F, B, C, D, E, H, L, peuvent être corrigés par déplacement du curseur sur la valeur affichée puis correction suivie de [Cr]. Pour sortir de l'instruction, taper [CR] sur une ligne vide.
C - Idem à "A" avec A', F', B', C', D', E', H', L'.
P - Idem à "A" avec PC, SP, IX, IY, I.
R - Affiche tous les registres du Z 80 sans correction possible.
X - Déplacement du contenu de la mémoire programme.
FROM? : Début de la zone à transférer.
To? : Fin de la zone à transférer.
TOP? : Début de la zone ou va aboutir le programme.

DIVERS
S - Enregistre le *Programme corrigé* sur K7.
FROM? : Adresse de début du programme actuellement en mémoir.
TO? : Adresse de fin du programme actuellement en mémoire.
Ce *programme corrigé* peut être chargé et utilisé comme un programme machine ; il comporte, contrairement au *Programme édité*, une table de symboles.
V - Vérifie le programme enregistré.
Y - Lit un programme qui peut être de 3 sortes :
1) *Programme corrigé* (avec "S")
2) *Programme édité* (voir EDITEUR DE LIENS)
3) *Programme machine* (écrit avec le prog "LANGAGE MACHINE"
- Liste du programme sur imprimante.
! - Retour au moniteur.
GOTO$1200 : Mémoire programme vide.
GOTO$1260 : Programme conservé.

Voici terminée la description "brute" des 3 programmes qUI composent L'ASSEMBLEUR. Mais, au fait, quels sont les avantages et les inconvénients de l'assembleur ?

Un programme écrit en assembleur :
— Tourne entre 10 et 20 fois plus vite que le même programme écrit en BASIC.
— Occupe environ 5 fois moins de place.
— Peut s'implanter sur les 48 K disponibles du MZ.
— Peut se composer de routines très rapides appelées par le BASIC (USR).

Les inconvénients :
— Nécessite une très bonne connaissance du microprocesseur Z 80 (structure interne, langage, instructions). Un excellent livre traite de ce sujet : "PROGRAMMATION DU Z 80" par Rodnay ZAKS aux Editions SIBEX.
– Demande de structurer et préparer soigneusement son programme avant de l'écrire. En effet, les corrections, si elles existent sont loin d'être aussi aisées qu'en BASIC.

QUELQUES BONNES ADRESSES

Afin de remplir vos longues soirées d'hiver, voici une liste de bonnes adresses (mémoires bien sur) du moniteur SP-1002 et du BASIC 5025. Vu la prolifération des BASICS, je me suis cantonné à ce dernier. Mais si vous trouvez les adresses correspondantes chez ses petits frères, ne vous génez pas pour nous les faire parvenir !

MUSIQUE

→ Tout d'abord, comment ne pas en faire : si vous mettez à 0 les octets 8764 et 8765, l'éxécution d'un programme BASIC ne reveillera plus vos voisins (retour à la normale par POKE 8764, 205 : POKE 8765, 48)

→ La routine de musique du moniteur est située à l'adresse $0030 (48). Si vous l'appelez MZ jouera les octets à partir de l'adresse contenue dans le registre DE. Le codage est le même qu'en BASIC. Le code de fin est $OD ou $C8. Si la carry est à 1 au retour, c'est qu'il y a eu un BREAK.

→ La routine située en $0041 (65) fixe le tempo. Celui-ci doit être mis dans le registre A.

→ Le tempo est stocké en complément à 8 en $119E (4510).

→ Si vous voulez directement contrôler les fréquences, la routine qui met en marche le synthétiseur est en $0044. La fréquence doit être mise en $11A2 et $11A1 (4514, 4515)

→ Pour arrêter le son la routine est en $0047 (71)

→ Au cas ou vous l'auriez oublié POKE 4509,0 ($1190) provoque un bip a chaque pression de touche.

CLAVIER

→ $45A4 (17828) contient l'ASC du dernier caractère entre par GET. Si avant chaque GET, vous remettez cet octet à zéro, plus besoin de pianoter : la touche sera prise en compte tant qu'elle sera enfoncée.

→ En $0003 est située la routine d'INPUT, les caractères entres sont stockes à l'adresse DE. Si l'on tape directement en BASIC : USR (3) CR CR puis ?1+1, l'effet est assez comique.

→ En $001B (27) on trouve une routine qui met dans l'accumulateur le code ASC de la touche enfoncée.

→ En $001E (30) routine qui teste si la touche SHIFT BREAK est enfoncée, si c'est le cas le blag Z est mis à 1 en retour. Si vous faites POKE 6639,24, la routine n'est plus appelée et vos programmes BASIC ne peuvent être interrompu.

→ En $09B3 (2483), routine d'attente de caractères avec clignotement du curseur. Au retour le code est dans le registre A.

→ Si l'octet $1170 (4464) est à 0 on passe en majuscule. Si il est à 1, on passe en minuscule.

CASSETTE

→ En $0021 (33), routine d'écriture de l'indicatif de fichier

→ En $0024 (36), routine d'écriture de fichier. Tout fichier doit être obligatoirement précédé d'un indicatif de fichier. Si par exemple, vous voulez copier le BASIC sur une autre cassette, vous devez taper tout de suite après l'avoir chargé : USR (33) ; USR (36).

→ En $0027 (39), routine de lecture d'indicatif de fichier.

→ En $002A (42), routine de lecture de fichier.

Pour ces quatre routines, les variables systèmes sont les mêmes :
— $10F0 (4336) : type de fichier (1 = MACHINE, 2 = BASIC)
— $10F1 à $1100 (4337 - 4353) : nom du fichier en ASC suivi de $OD
— $1102 & $1103 (4354 - 4355) : longueur en octets
— $1104 & $1105 (4356 - 4357) : adresse du premier octet du fichier
— $1106 & $1107 (4358 - 4359) : adresse d'éxécution

→ Avant un SAVE, POKE 10680, 1 protége le programme en empêchant les listings et les copies. POKE 10682, 1 fait démarer les programmes automatiquement.

ECRAN

→ La RAM VIDEO est situé de $D000 à $D3E8, entrée libre par PEEK et POKE.

→ En $E102 (57602), contrôle du tube cathodique et de la LED :
Si l'octet est à 5, l'écran est allumé et la diode verte
Si l'octet est à 4, l'écran est éteint et la diode verte
Si l'octet est à 1, l'écran est allumé et la diode rouge
Si l'octet est à 2, l'écran est éteint et la diode rouge

→ En $0006, routine de saut de ligne.

→ En $0012 (18), routine d'affichage d'un caractère, l'ASCII doit être dans le registre A.

→ En $0015 (21), routine qui affiche les caractères ASCII stockes à l'adresse DE. Le code de fin de message est $OD.

→ En $0DDC (3548), routine chargée des effets spéciaux. Elle agit en fonction du contenu de l'accumulateur ($C1 = ↓, $C2 = ↑, $C3 = →, $C4 = ←, $C5 = home, $C6 = CLR, $C7 = DEL, $C8 = INS, $C9 = SCROLL, $CA = SML, $CB = CR)

→ L'abcisse du curseur est en $1171 (4465) et l'ordonnée en $1172 (4466)

DIVERS

→ L'ordinateur se branche en $1038 (4152) en cas l'interuption (par exemple lorsque l'horloge passe de 12 h 59 min. 59 s. à 1 h 00 00)

→ En $0033 (51), routine de mise à l'heure. Celle-ci doit être mise en secondes dans le registre DE ; et l'accumulateur doit être à 1 si l'on est dans l'après-midi.

→ En $003B (59), routine de lecture de l'heure, même système.

→ En $03DA (986), routine qui convertit en ASCII le chiffre contenu dans A. Le résultat est rangé dans A.

→ $03F9 (1017) : routine inverse.

→ En $0410 (1040), routine qui met dans le registre HL le nombre hexa décimal stocké en ASCII à l'adresse DE.

→ La routine située en $041F (1055) fonctionne de la même façon mais le résultat est rangé dans l'accumulateur.

→ (3001), routine qui convertit A d'ASCII en code d'affichage.

→ $OBCE (3022) : le contraire.

→ Si l'octet $27B7 (10167) est différent de 0, le PEEK peut aller fouiner ou il veut sans retourner 32 dans les zones protégées.

Simon CHAGNOUX.

Quand l'équipe SOFT s'en mêle : Michel STERMANN est analyste programmeur au sein de l'équipe SOFT de SBM. Nous avons pu lui soutirer ce petit programme pour MZ 80 A, qui permet de transferer le moniteur en RAM, et de transformer le clavier alphabétique en AZERTY.

LE FOUINEUR

```
## PROGRAMME AZERTY ## AVR.81 ##
## Michel STERMANN  ## S.B.M. ##
1 REM ###################################
2 REM ###################################
3 REM ###################################
100 PRINT"■":GOSUB 10000
110 PRINT"J'AI TRANSFERE LE MONITEUR EN RAM"
120 POKE$0BED,$01:REM Q←→A
130 POKE$0BEE,$11
140 POKE$0C35,$81:REM q←→a
150 POKE$0C36,$91
160 POKE$0C6E,$3C:REM _←→♥
170 POKE$0C6D,$53
180 POKE$0CA5,$46:REM ♠←→_
190 POKE$0CA6,$7C
200 POKE$0CDE,$C1:REM ^Q←→^A
210 POKE$0CDD,$E0
220 REM ---
230 POKE$0BF1,$17:REM Z←→W
240 POKE$0BEC.$1A
250 POKE$0C39,$97:REM z←→w
260 POKE$0C34,$9A
270 POKE$0C71,$38:REM /←→_
280 POKE$0C6C,$76
290 POKE$0CA9,$78:REM \←→-
300 POKE$0CA4,$77
310 POKE$0CE1,$E2:REM ^Z←→^W
320 POKE$0CDC,$E5
330 REM ---
340 POKE$0C11,$2C:REM ;←→M
350 POKE$0C17,$0D
360 POKE$0C5F,$8D:REM +←→m
370 POKE$0C59,$6A
380 POKE$0C97,$1C:REM ╄←→◥
390 POKE$0C91,$42
400 POKE$0CCF,$1D:REM ┘←→\
410 POKE$0CC9,$B5
800 PRINT"CLAVIER AZERTY"
900 LIMIT $BFFF:NEW
1000 I=I+1:P=PEEK(I)
1010 PRINT I;P,
1020 IF P<32 THEN P=112
1030 PRINT CHR$(P)
1050 GET Y$:IF Y$=" " GOTO 1050
1060 GOTO 1000
10000 LIMIT $BF00
10010 POKE $BF00,$3A :REM LD A,(E00C)
10020 POKE $BF01,$0C
10030 POKE $BF02,$E0 :REM <> E010
10040 POKE $BF03,$21 :REM LD HL.C000
10050 POKE $BF04,$00
10060 POKE $BF05,$C0
10070 POKE $BF06,$11 :REM LD DE,0000
10080 POKE $BF07,$00
10090 POKE $BF08,$00
10100 POKE $BF09,$01 :REM LD BC,1000
10110 POKE $BF0A,$00
10120 POKE $BF0B,$10
10130 POKE $BF0C,$ED :REM LDIR
10140 POKE $BF0D,$B0
10150 POKE $BF0E,$C9 :REM RET
10160 USR($BF00)
10900 RETURN
```

MEA CULPA

Une grosse erreur s'est glissée dans le programme de copie du BASIC ; il fallait bien lire en ligne 60 GETA$: IFA$ = ""THEN60.

Ce programme ne peut tourner qu'avec 64K de ram puisqu'il va copier le BASIC à partir de l'adresse $8000, 1ère adresse de la RAM 32K optionnelle.
Autre omission ; elle concerne le programme de répétition automatique de caractères qui ne pouvait tourner qu'avec 64K de MEV. Voici ce même programme adapté aux possesseurs de MZ équipés de 32K.
10 FORI = 1TO21 : READA : POKE 32735 + I, A : NEXTI
20 FORI = 1TO03 : READA : POKE 2440 + I, A NEXTI
30 END
40 DATA 237, 91, 3B, 0, 120, 254, 9, 192, 237, 91, 6, 0, 6, 0, 197, 1, B0, 0, 195, 4, 5
50 DATE 205, 224, 127

QUELQUES ADRESSES UTILES

$0947 POKE $0947, $05 permet de modifier les valeurs de scrolling (CONSOLE S LIGNESUP, LIGNEINF) sans effacer le contenu de l'écran.
$1C10 POKE $1C10, $C9: POKE $1C19, $C9 autorise l'insertion d'instructions 'directes' (ex : LIST) dans un programme.
$00AA POKE $00AA, 0 produit un "BIP" à chaque frappe de touche. POKE $00AA, FF rend votre MZ à nouveau muet.
$06DD Vitesse de clignotement du curseur. < $40 : + lent.
$0725 Vitesse de répétition du déplacement du curseur (normal = $40).
$0897 POKE $897, 0 fait monter le scrolling d'une seule ligne.
$097C POKE $097C, 0 supprime le contrôle anti-rebond du clavier, d'ou répétition automatique des caractères. Cette répétition, très rapide peut être utilisée dans l'élaboration de jeux intéractifs au clavier
$1180-121F : Contenu des touches F1 à F10.

(N.B.) Il existe au moins 5 versions de BASIC 5510 et 3 versions de 6510 ; toutes ces versions sont compatibles au niveau logiciel mais pas du tout au niveau implantation mémoire. Il est donc fort probable qu'une modification mémoire fonctionne très bien sous une certaine version de BASIC et pas sous une autre. Nous sommes, par contre tranquilles avec le MONITEUR SB 1510 qui à l'air universel.

QUELQUES CODES ASCII MECONNUS
CHR$ (11) = [BREAK]
CHR$ (13) = [CR] et [ENT]
CHR$ (14) = [GRPH] et [SFTLOCK] et [CANCEL]
CHR$ (16) = [F1], CHR$ (17) = [F2], ..., CHR$ (25) = [F10]
CHR$ (26) = [00]
CHR$ (27) = [TAB]
Il ne nous manque plus que les codes 28, 29 et 30...

AS-TU CES ASTUCES ?

(1) **- Passer sous moniteur
- Modifier en $239B : E3, 73, 23, 72, 00, 00, 00
- Vous pouvez poker maintenant 2 octets en même temps.
ex : POKE $E000, $FF01 donne ($E000)=01, ($E001)=FF. Très pratique lorsqu'on manipule des valeurs de 16 bits.
(2) 00 N'importe quel argument peut être placé après CHR$
ex : ?CHR$ (40896/852) donnera "0".
(3) 00 Ce petit sous-programme vous permettra d'afficher toutes vos valeurs numériques avec marge à droite et 2 décimales.
100 INPUT "Votre somme en francs. centimes :" A
100 GOSUB 9000
120 PRINT TAB (10) ; A ; TAB [40-LEN (A$)] ; A$
130 END
9000 A = INT (A*100)/100 : A$ = STR$ (A) : IF (LEN (A$) = 2) GOTO 0930
9010 P$ = RIGHT$ (A$,3) : IFLEFT$ (P$,1) = "."THEN RETURN
9020 IF MID$ (P$, 2, 1) = "."THENA$ = A$ + "0" : RETURN
9030 A$ = A$ ".00" : RETURN
9030 A$ = A$ + ".00" : RETURN

ex : O	0.00
0.9	0.90
999	999.00
894530,34	8945230,34

(4) **Autre sous-programme qui vous permettra d'arrondir un réel quelconque au nombre de décimales voulu.
2040 INPUT "REEL A ARRONDIR..?" ; RE
2050 INPUT "NOMBRE DE DECIMALES A RETENIR..?" ; PP
2060 KK = RE ;GOSUB 2090
2070 ?"LE RELLE.." ; RE ; "DEVIENT...", KK :GOTO2000
2090 P = PP + 1
2100 IF INT (KK*10P̂) - 10*INT [K*10(P-1)]>5 THEN 2120
2110 KK = [INT (KK*10(P-1))]/10(P-1) : GOTO 2130
2120 KK = (INT (KK*10(P-1))+1)/10 ↑ (P-1)
2130 RETURN

(5) ** Les quelques lignes qui suivent permettent aux possesseurs de SEIKOSHA GP 100 d'effectuer une copie graphique de l'écran de leur MZ. Cest possible à condition d'avoir introduit, au préalable, les modifications d'adresse mémoire qui permettent d'utiliser l'imprimante avec ses ordres clés d'origine. 2 possibilités d'utiliser ce programme :
1) L'introduire à la fin de votre programme graphique et faire RUN 9000.
2) Le charger et le faire tourner après avoir mis en mémoire graphique le dessin à copier.
9000 FORA = 1T0319
9010 FORB = 0T06
9025 IFPOINT (A, B, +C) = 3 THENN = N + 10^B
9030 NEXTB
9050 IFN = OTHENH = O : GOTO9060
9055 GOSUB90200
9060 PRINT/PCHR$ ($B) + CHR$ (H + 128) ;
9070 NEXTA
9080 PRINT/PCHR$ (1)
9090 IFC>192THENEND
9100 C=C+7
9110 GOTO9000
9200 Q = 0 : H = 0
9210 M = 10*INT (N/10)
9220 P = (N-M)*(2Q) : N = M/10 : H = H + P : Q = Q + 1
9230 IFN = OTHENH = INT (H+.2) : RETURN
9240 GOTO9210

(6) ** Cette routine en assembleur permet l'écriture directe sur la RAM écran basse résolution du 80B. Pour mémoire (*), cette RAM va de $5000 à $RFFF avec 32K de MEV, ou de $D000 avec 64K de MEV.

— Passez sous moniteur et entrez cette suite d'instruction avec l'instruction "M"

```
C000 F5 : PUSH AF
COO1 E5 : PUSH HL
COO2 DB : IN A. (E8H)
COO3 EB
COO4 CB : SET 7,A
COO5 FF
COO6 D3 : OUT (E8H),A
C007 E8
COOB 2A : LD HL, (CO20H)
COO9 20
C001 CO
COOB 3A : LD A, (CO22H)
COOC 22
COOD CO
CCOE 77 : LD (HL,A


COOF DB : IN A,(EBH)
C100 EB
C101 CB : RES 7,A
C1O2 BF
C103 D3 : OUT (EBH),A
C104 EB
C105 E1 : POP HL
C106 F1 : POP AF
C107 C9 : RET
```

Repasser ensuite sous BASIC sans effacer la mémoire (J $1280).

10 POKE $C020, $10 : REM adresse basse écran ($DO10, au hasard)
20 POKE $CO21, $DO : REM adresse haute écran
30 POKE $C022, $31 : REM Code ASCII du caractère à afficher.

Il est également possible, pour gérer cette mêmê RAM vidéo, d'appeler les routines-moniteur suivantes :

$OC68 DSPR
ENTREE : HL = Adresse RAM VIDEO à lire.
SORTIE : A = Code ASCII du caractère lu.

$OC71 DSPW
ENTREE : HL = Adresse RAM VIDEO ou écrire
A = Caractère à écrire.
SORTIE : Affichage du caractère sur l'écran.

$0C87 DWLDIR
ENTREE : HL = Adresse RAM VIDEO ou écrire.
DE = Adresse du début de la chaîne à écrire.
BC = Longueur de cette chaîne
SORTIE : Affichage de la chaîne sur l'écran, au bon endroit

$0C56 PNT1
ENTREE : HL = Coordonnées X, Y.
SORTIE : HL = Adresse de la RAM VIDEO correspondant a X et Y.

$OC96 DSCL
ENTREE : HL = Adresse de début de la RAM VIDEO a effacer.
B = Nombre de caractères (<255) a effacer.
SORTIE : Effacement de B caractères sur l'écran, au bon endroit.

Nous n'avons encore, malheureusement, pratiquement aucun renseignement concernant les possibilités d'attaquer directement les RAM graphiques. Toute découverte faite dans ce domaine sera la bienvenue au Club.

Soyez logique ! Ou de l'utilisation des fonctions logiques.

Les fonctions logiques semblent trop souvent méconnues dans les programmes, pourtant elles peuvent être d'une grande utilité (clarté, simplification et rapidité d'exécution).

Tout d'abord quelques rappels : un opérateur de comparaison (= > < > = < = < >), placé entre deux variables ou constantes donne un résultat simple : 0 si la proposition est fausse, — 1 si elle est vraie. C'est ainsi que ? 4=4 donne — 1, ? 4= 5 donne 0.

Si A= 2 ? A = 2 donne — 1 et ? A = 3 donne 0.

Le résultat (tout comme le résultat d'une opération arithmétique) peut être utilisé dans différents calculs (ne pas oublier les parenthèses, les opérateurs logiques n'ayant aucune priorité).

Afin de n'être pas trop théorique nous vous proposons quelques exemples, le champ d'applications étant très grand, vous pourrez selon vos besoins en découvrir bien d'autres.

AFFICHAGE :
Si on veut afficher le score d'un jeu ; ? N ; « POINT » ; CHR $ (—115 * (N> 1)) permet en une ligne d'éviter une faute d'orthographe.

AFFICHAGE PARAMETRE :
Sur un damier un ∅ représente une case vide, 1 le pion blanc, 2 le pion noir (V = 0 1 ou 2), on peut donc afficher le caractère correspondant par :
CHR $ (—32 * (V= 0)— 148 * (V= 1) — 147 * (V= 2).

DEPLACEMENT D'UN POINT DANS UN TABLEAU :
La variable de déplacement (V) prend par exemple la valeur 1 si le point se déplace vers le haut, 2 vers la droite, 3 vers le bas et 4 vers la gauche.
On peut remplacer les quatre IF V= THEN nécessaires pour le calcul des nouvelles coordonnées pour X= X — (V=)2 + (V= 4) : Y= Y — (V= 3) + (V=1) et le tour est joué !

SAISIE D'UNE REPONSE (O/N) :
10 GET R $: ON 1 — (R$= « 0 ») — 2 * (R S = « N ») GOTO 10, 20, 30.

SI R$= « 0 » branchement en 20, si R $ = « N » en 30 sinon en 10, élégant n'est-ce pas ?

BRANCHEMENT AVEC TESTS MULTIPLES :
Dans le même ordre d'idée lorsque le branchement s'effectue par l'intermédiaire de la variable R on aura :

ON 1 — (R= 1) — 2 * (R= 2) — 3 * (R= 3) GOTO 100, 200, 300, 400.

FLAG DE PARITE :
Si à l'intérieur d'une boucle le flag F doit prendre alternativement la valeur 0 ou 1 on pourra écrire : F = — (F= 0). L'essayer c'est l'adopter !

En conclusion, la réflexion apportée sur ces tests logiques vous permettra de récupérer quelques octets dans les programmes µS au passage. Bon courage.

A propos, si l'imperturbable pavé clignotant vous fatigue, changez-le pour un autre signe, et pourquoi pas par une flèche (code ASCII 131) grâce aux deux POKE suivants :

POKE $ 06C8, 131 et POKE $ 06D7, 131.

De même, dans la grande série « simplifiez vos programmes », USR ($ 0517) permettra une attente d'environ 2 s et USR ($ 0500) de 4 s en remplacement des boucles FOR I= to 3000 : NEXT où rien ne se passe.

M. MILLET MAUREPAS.

Enquête : Les Sharpentiers et leur club

Pour nous permettre de mieux connaître vos goûts et vos besoins et de ce fait améliorer les relations du club avec ses membres nous vous remercions de bien vouloir nous retourner ce questionnaire dûment rempli.

LE SHARPENTIER

Nom :

Prénom :

Adresse :

Profession :
Age :

Combien possédez-vous de matériels SHARP et lesquels ?

LE BULLETIN

– Le bulletin vous satisfait-il dans sa nouvelle présentation ?

– Quels sont les rubriques qui vous intéressent le plus ?

– Est-il trop technique
Pas assez technique ?

– Manque-t-il d'articles d'intérêt général. Si oui de quel type ?

– Désirez-vous voir développer une rubrique Produits SHARP plus importante (pas spécialement la micro) ?

– Désirez-vous des photos, des dessins, des exemples ?

– Des petites annonces ?

– Vos suggestions ?

LE CLUB

– Le fonctionnement du CLUB vous satisfait-il ?

– Que faire pour l'améliorer ?

– Accepteriez-vous une participation financière (cotisation) permettant de recevoir le bulletin et d'acquérir des programmes ?

– Pourriez-vous participer activement à la vie du club ?

– Etes-vous intéressé par l'idée d'une réunion générale des SHARPENTIERS à PARIS ?

– Pourriez-vous être animateur d'un club régional ?

– Le correspondant du club en province ?

– Ou seriez-vous intéressé par l'intégration des Sharpentiers à des Clubs organisés du type microtel pour nous permettre une bonne couverture nationale ?

Vos suggestions :

Date : Signature :

MZ 80 K
CITY

```
5 TEMPO 7:MUSIC "R0":POKE 10407,0
10 GOSUB 3000
20 FOR I=54169 TO 54206
30 FOR J=I-40*INT(T*RND(1)) TO I        STEP +40
40 POKE J,255
50 NEXT J,I
60 FORI=54208TO54248:POKEI,166:NEXTI
100 FOR I=53248 TO 54207
110 IF PEEK(I+1)=255 GOTO 2000
120 POKE I,0
130 POKE I+1,199
135 IF F>4 GOTO 1000
140 FORK=0TO99:NEXTK
145 GET X$:IF X$=" " THEN X=I+40:F=5:USR(62)
150 NEXT I
160 PRINT "▯▯▯▯▯▯▯▯▯▯▯▯Bravo !!!":GOTO 2000
1000 T=PEEK(X):POKE X,88
1005 FORK=0TO69:NEXTK
1010 IF X>54207 THEN F=0:MUSIC "AB":POKEX,166:GOTO 145
1020 IFT=255THENPOKEX,0:F=INT(RND(1)*10):MUSIC"CB":Z=Z+1:GOTO145
1030 POKEX,0:X=X+40:GOTO 150
2000 MUSIC "¯C¯D¯E¯F¯G¯B¯ACDEFGAB_C_D_E_F_G_A_B"
2005 FORP=0TO10:POKE57347,0
2010 FORG=0TO99:NEXTG:POKE57347,1
2015 FORG=0TO99:NEXTG,P
2020 PRINT "▯▯▯▯▯▯▯▯▯▯▯▯▯▯Votre score :";Z
2025 INPUT"▯▯▯▯▯▯Une autre partie : ";R$:IFR$="O"THENCLR:GOSUB 3120:GOTO 20
2030 POKE 10407,184:PRINT "▯"
2040 END
3000 PRINT "▯▯▯▯▯▯"
3010 PRINT TAB(10);"▯▯▯▯ ▯▯▯ ▯▯▯▯▯ ▯▯ ▯▯"
3020 PRINT TAB(10);"▯      ▯     ▯     ▯▯▯ "
3030 PRINT TAB(10);"▯      ▯     ▯      ▯  "
3040 PRINT TAB(10);"▯▯▯▯ ▯▯▯    ▯      ▯  "
3045 FOR I=0 TO 9
3050 PRINT "▯▯▯▯▯"
3060 PRINT TAB(16);"●"
3070 FOR J=0 TO 99
3075 NEXT J
3080 PRINT "▯▯▯▯▯"
3090 PRINT TAB(16);CHR$(103)
3100 FOR J=0 TO 99
3110 NEXT J,I
3120 MUSIC "CDAFECDBCFGDBGDFCGBDGCDGGC"
3130 PRINT"▯▯▯▯▯▯▯▯▯▯▯▯▯▯▯▯▯▯▯▯Niveau de difficulte▯▯,▯ (1-3)";SPC(48)
3140 PRINT"▯▯▯▯▯▯▯▯▯▯▯▯"
3150 PRINT TAB(32);:INPUT " : ";T
3160 IF (T<1)+(T>3) THEN 3140
3170 T=T*5:PRINT "▯":RETURN
3240 PRINT TAB(25);INPUT " : ";T
```

MZ 80 K
COURSE

```
1 REM ** 48K RAM INDISPENSABLES **
3 DATA 205,27,0,50,194,195,201
4 FORI=53100TO53106:READA:POKEI,A:NEXT
5 DATA-1:MUSIC"A0"
10 REM COURSE DE VOITURES
20 PRINT"▒▒▒":E=54048
30 PRINTTAB(6);"▒▒▒ ▒▒▒ ▒  ▒ ▒▒▒ ▒▒▒ ▒▒▒"
40 PRINTTAB(6);"▒   ▒ ▒ ▒  ▒ ▒ ▒ ▒   ▒"
50 PRINTTAB(6);"▒   ▒ ▒ ▒  ▒ ▒▒▒ ▒▒▒ ▒▒▒"
60 PRINTTAB(6);"▒   ▒ ▒ ▒▒▒ ▒ ▒   ▒ ▒"
70 PRINTTAB(6);"▒▒▒ ▒▒▒ ▒▒▒ ▒  ▒ ▒▒▒ ▒▒▒"
72 PRINT:PRINT:PRINT:PRINT
75 GOSUB20000
80 V=3:GOSUB5000
85 TI$="000000"
86 GETR$:IFR$=""THEN86
90 FL=0:PRINT"▒"
100 REM INITIALISATION ROUTE VOITURE
110 D=0:X=20
120 FORI=0TO24
130 POKE53248+40*I+12,255:POKE53248+40*I+28,255
140 NEXTI
150 POKEE+X,201
160 REM  CHOIX ALEATOIRE DE LA ROUTE
170 Z=1+INT(9*RND(7))
180 ONZGOTO200,300,400,700,800,900,1000,1100,1200
200 REM DROITE:A=0
210 FORI=1TO25
220 POKE53260,255:POKE53276,255
230 GOSUB4000:IFFL=1THEN90
240 NEXTI
250 GOTO170
300 REM VIRAGES
310 IN=SGN(.5-RND(1))*π/16:Y=0
320 FORI=1TO96
330 A=INT(10*SIN(Y))
340 POKE53248+12+A,255:POKE53248+28+A,255
345 POKE53248+13+A,255:POKE53248+27+A,255
350 GOSUB4000:IFFL=1THEN90
360 Y=Y+IN
370 NEXTI
380 GOTO170
400 REM  CHICANES
410 FORI=1TO10:POKE53248+12,255:POKE53248+28,255
420 GOSUB4000:IFFL=1THEN90
430 NEXTI
435 PRINT"▒";
440 PRINTTAB(12);"xxxxxxxxxxx      x"
450 GOSUB4000:IFFL=1THEN90
460 FORI=1TO10:POKE53248+22,255:POKE53248+28,255
470 GOSUB4000:IFFL=1THEN90
480 NEXTI
485 PRINT"▒";
490 PRINTTAB(12);"xxxxxxxxxxx      x"
500 GOSUB4000:IFFL=1THEN90
510 FORI=1TO10:POKE53248+12,255:POKE53248+28,255
520 GOSUB4000:IFFL=1THEN90
530 NEXTI
535 PRINT"▒";
```

```
540 PRINTTAB(12);"×     ××××××××××××"
550 GOSUB4000:IFFL=1THEN90
560 FORI=1TO5:POKE53248+12,255:POKE53248+17,255
570 GOSUB4000:IFFL=1THEN90
580 NEXTI
585 PRINT"◘";
590 PRINTTAB(12);"×     ××××××××××××"
600 GOSUB4000:IFFL=1THEN90
610 FORI=1TO10:POKE53248+12,255:POKE53248+28,255
620 GOSUB4000:IFFL=1THEN90
630 NEXTI
640 GOTO170
700 REM  ROUTE GOGOLE
710 FORI=1TO50
720 POKE53248+12,255:POKE53248+28,255
730 GOSUB4000:IFFL=1THEN90
740 H=INT(16*RND(2))
750 POKE53248+H+12,166
760 POKE53248+12,255:POKE53248+28,255
770 GOSUB4000:IFFL=1THEN90
780 NEXTI
790 GOTO170
800 REM LIGNE
810 PRINT"◘                    ▙▀▄▀▄▀▜   ▛▄▀▄▀"
820 POKE53248+12,255:POKE53248+28,255
825 GOSUB4000
830 IFRND(66)>0.4GOTO810
840 GOTO170
900 O=13+INT(RND(π)*9)
901 POKE53248+12,255:POKE53248+28,255
910 PRINT"◘";TAB(O);" ▀▀▀"
915 GOSUB990
920 PRINTTAB(O);" ◣ ◢"
925 GOSUB990
930 PRINTTAB(O);"×▓▓▓×"
935 GOSUB990
940 PRINTTAB(O);" █▓▓█ "
945 GOSUB990
950 PRINTTAB(O);" █▓▓█ "
955 GOSUB990
960 PRINTTAB(O);"×▓▓▓×"
965 GOSUB990
970 PRINTTAB(O);"×▓▓▓×"
975 GOSUB990
980 PRINTTAB(O);" ▁▁▁▁"
985 GOSUB990
986 GOTO170
990 PRINT"◘";:GOSUB4000:POKE53260,255:POKE53276,255:RETURN
1000 FORI=1TO8:POKE53248+12,255:POKE53248+28,255:GOSUB4000:NEXT
1005 PRINT"◘                 ░░░░░░░░░░░░░░░░░"
1010 POKE53248+12,255:POKE53248+28,255
1020 POKE53261+RND(54)*15,0:GOSUB4000:IFRND(777)<.8GOTO1005
1025 FORI=1TO7:POKE53248+12,255:POKE53248+28,255:GOSUB4000:NEXT
1030 GOTO170
1100 O=16:FORI=12TO19:POKE53248+I,255:POKE53248+I+O,255:GOSUB4000:O=O-2:NEXT
1110 PRINT"◘▫▫▫▫▫▫▫▫▫▫▫▫▫▫▫▫▫▫▫▫▫   ▫▫▫▫▫▫▫▫▫▫▫▫▫▫▫▫▫▫▫▫▫"
1120 O=2:FORI=19TO12STEP-1:POKE53248+I,255:POKE53248+I+O,255:GOSUB4000:O=O+2
1130 NEXT
1140 PRINT"◘▫▫▫▫▫▫▫▫▫▫▫▫▫▫▫▫               ▫▫▫▫▫▫▫▫▫▫▫▫▫▫▫"
1150 GOTO170
1200 O=16:FORI=12TO18:POKE53248+I,255:POKE53248+I+O,255:GOSUB4000:O=O-2:NEXT
1210 FORJ=1TORND(88)*50
1220 FORI=1TORND(67)*10
1230 POKE53266,255:POKE53270,255:GOSUB4000:NEXT:POKE53267+INT(RND(12)*3),201
1240 NEXT
1250 O=4:FORI=18TO12STEP-1:POKE53248+I,255:POKE53248+I+O,255:GOSUB4000:O=O+2
```

```
1260 NEXT:GOTO170
4000 REM  MOUVEMENT VOITURE
4010 USR(53100):R$=CHR$(PEEK(50114)):D=0
4020 IFR$="*"THEND=1
4030 IFR$=" "THEND=-1
4040 POKEE+X,0:X=X+D
4050 USR(32768)
4060 IF(PEEK(X+E)<>0)THEN4080
4070 POKEE+X,201:RETURN
4080 REM VOITURE PERDUE:FORK=0TO5
4090 POKEE+X,91:TEMPO7:MUSIC"1BC":POKEE+X,108:MUSIC"DE"
4100 NEXTK
4110 GOTO4200
4140 FL=1
4150 GOTO4070
4200 REM   FIN DU JEU
4210 A$=TI$
4220 IFVAL(A$)>VAL(SC$)THENSC$=A$
4230 PRINT"■■■■■"
4240 PRINT"Votre score est :";A$
4250 PRINT"■■■Le record est    :";SC$
4260 PRINT"■■■■■■■Tapez une touche pour rejouer"
4270 R$=""
4280 GETR$:IFR$=""THEN4280
4290 GOTO10
5000 LIMIT32767:RESTORE5040
5010 FORI=32768TO32802
5020 READD:POKEI,D
5030 NEXTI:RETURN
5040 DATA33,191,211
5050 DATA17,231,211
5060 DATA1,192,3
5070 DATA126
5080 DATA18
5090 DATA43
5100 DATA27
5110 DATA11
5120 DATA120
5130 DATA177
5140 DATA194,9,128
5150 DATA33,0,208
5160 DATA1,40,0
5170 DATA54,0
5180 DATA35
5190 DATA11
5200 DATA120
5210 DATA177
5220 DATA194,25,128
5230 DATA201
20000 PRINT"         ▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒"
20002 PRINT"         ▌                       ▌"
20004 PRINT"         ▌      DEPLACEMENT      ▌"
20006 PRINT"         ▌                       ▌"
20008 PRINT"         ▌Gauche           Droite▌"
20010 PRINT"         ▌                       ▌"
20012 PRINT"         ▌ SP               CR   ▌"
20014 PRINT"         ▌                       ▌"
20016 PRINT"         ▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒"
20018 RETURN
```

MZ 80 B
LABY


```
 0 REM************ LABY **************
 0 REM************ Jean Millet *******
 0 REM************ SEPT. 82 **********
 10 GOSUB60000:POSITION55,54:FORI=1TO13:A$="":FORJ=1TO32:READB:A$=A$+CHR$(B):NEX
TJ:PATTERN-16,A$:NEXTI:FORI=1TO8:BLINE55,54-I,263,54-I:NEXT
 20 DATA0,3,6,6,6,6,3,1,0,0,9,21,19,96,0,0,224,16,48,48,48,48,24,0,128,128,0,8,1
52,240,96,0
 22 DATA3,4,8,24,25,25,24,26,26,27,25,8,4,3,1,0,4,216,112,128,0,16,216,96,48 ,48
,164,204,24, 48,224,0
 24 DATA0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0
 26 DATA0,3,6,6,6,6,3,1,0,0,9,21,19,96,0,0,224,16,48,48,48,48,24,0,128,128,0,8,1
52,240,96,0
 28 DATA8,21,50,54,54,22,14,50,36,8,24,24,24,12,3,0,24,32,160,96,96,96,96,96,96,
96,96,104,100,100,248,0
 30 DATA2,5,13,13,13,5,9,1,1,2,4,12,22,35,1,0,32,80,152,152,136,132,216,152,12,1
2,12,12,8,16,224,0
 32 DATA6,9,12,12,6,12,8,17,0,0,0,0,12,6,3,0,48,72,204,198,198,198,134,6,134,76,
56,48,96,192,128,0
 34 DATA4,10,19,51,27,15,11,83,35,3,27,39,35,22,12,0,32,80,136,12,12,12,8,16,224
,96,48,52,50,26,12,0
 36 DATA12,19,17,8,4,0,0,1,1,17,40,12,6,6,3,0,16,8,136,240,0,96,192,128,128,128,
192,96,48,48,224,0
 38 DATA6,9,9,13,5,21,9,1,1,1,1,9,17,11,6,0,112,72,140,140,140,140,140,136,152,1
52,152,144,146,10,6,0
 40 DATA7,11,16,18,12,0,1,1,0,0,4,8,7,15,16,0,226,228,56,32,64,128,128,128,128,9
6,96,64,136,240,224,0
 42 DATA6,11,17,8,4,5,21,38,36,24,3,4,4,2,1,0,16,32,192,0,224,48,24,24,24,24,24,
144,96,16,224,0
 44 DATA3,4,8,24,25,25,24,26,26,27,25,8,4,3,1,0,4,216,112,128,0,16,216,96,48 ,48
,164,204,24, 48,224,0
 100 DIMT(21,15),XG(10),XD(10),YH(10),YB(10),D$(5):V=1:PV=200:PM=50:MM=4:FC=0
 110 FORI=0TO10:READXG(I):XD(I)=319-XG(I):YH(I)=XG(I)/1.6:YB(I)=199-XG(I)/1.6:NE
XT:DATA0,15,55,85,110,130,145,153,158,160,160
 120 D$(0)="     0     ":D$(1)=" 0       0 ":D$(2)=" 0   0   0 ":D$(3)="0 0     0 0":D$(
4)="0 0 0 0 0":D$(5)="000    000"
 130 FORI=0TO21:T(I,0)=9:T(I,15)=9:NEXT:FORI=1TO14:T(0,I)=9:T(21,I)=9:NEXT
 140 GOSUB5000:FORJ=1TO7:FORI=1TO10:READT(I,J):NEXTI,J:GOSUB5000:FORJ=1TO7:FORI=
11TO20:READT(I,J):NEXTI,J:GOSUB5000:FORJ=8TO14:FORI=1TO10:READT(I,J):NEXTI,J
 150 GOSUB5000:FORJ=8TO14:FORI=11TO20:READT(I,J):NEXTI,J:I1=4:J1=14:T1=10:T(4,14
)=1:IP=11:JP=1:GOSUB200:T(I,J)=2:GOSUB200:T(I,J)=4:IP=1:JP=1:GOSUB200:T(I,J)=3
 160 I3=I:J3=J:GOSUB200:T(I,J)=4:GOSUB200:T(I,J)=5:IP=11:JP=8:GOSUB200:T(I,J)=5:
GOSUB200:T(I,J)=4:IP=1:JP=8:GOSUB200:T(I,J)=5:GOSUB200:T(I,J)=4:GOTO205
 200 I=INT(10*RND(1)+IP):J=INT(7*RND(1)+JP):IFT(I,J)<>0THEN200
 202 RETURN
 205 CURSOR7,22:PRINTSTRING$("■",27):PRINTTAB(7);"     MODE D'EMPLOI? (O/N)     ":P
RINTTAB(7);STRING$("■",27)
 210 GETR$:ON1-(R$="O")-2*(R$="N")GOTO210,50000,220
 220 CONSOLEN:GOSUB10000:CURSOR6,2:PRINT"VUE GENERALE DU LABYRINTHE...":GOTO810
 300 IFPV<0THEN30000
 305 NV=0:NP=0:NT=0:NS=0:N=0:FD=0:FG=0:GRAPH I1,C,O1:PRINTCHR$(6):CURSOR12,0:PRIN
T"Points de vie:";PV;" ";CHR$(-129*(V=3)-130*(V=1)-131*(V=2)-132*(V=4))
 310 I=I1+N*(V=4)-N*(V=2):J=J1+N*(V=1)-N*(V=3):CG=-T(I-1,J)*(V=1)-T(I,J-1)*(V=2)
-T(I+1,J)*(V=3)-T(I,J+1)*(V=4)
 320 CD=-T(I+1,J)*(V=1)-T(I,J+1)*(V=2)-T(I-1,J)*(V=3)-T(I,J-1)*(V=4):CF=-T(I,J-1
)*(V=1)-T(I+1,J)*(V=2)-T(I,J+1)*(V=3)-T(I-1,J)*(V=4):IFCG=9THEN380
 370 IFFG=0THENFG=1:NG=N
 375 GOTO400
 380 IFFG=1THENLINEXG(NG),YH(N),XG(N),YH(N),XG(N),YB(N),XG(NG),YB(N):FG=0
 390 LINEXG(N),YH(N),XG(N+1),YH(N+1),XG(N+1),YB(N+1),XG(N),YB(N):NG=N+1
 400 IFCD=9THEN420
 410 IFFD=0THENFD=1:ND=N
 415 GOTO440
 420 IFFD=1THENLINEXD(ND),YH(N),XD(N),YH(N),XD(N),YB(N),XD(ND),YB(N):FD=0
 430 LINEXD(N),YH(N),XD(N+1),YH(N+1),XD(N+1),YB(N+1),XD(N),YB(N):ND=N+1
 440 IFCF=9THENLINEXG(NG),YH(N+1),XD(ND),YH(N+1):LINEXG(NG),YB(N+1),XD(ND),YB(N+
1):GOTO480
 450 N=N+1:IF(CF=3)*(N<6)*(F=0)THEN1000
 455 IFCF=5THENNP=N
 457 IFCF=4THENNV=N
 460 IFCF=2THENNT=N
 465 IFCF=10THENNS=N
 470 IFN<10THEN310
 480 IF(NT=0)+(NT>5)THEN550
```

```
490 FORI=1TO6-NT:LINE157+NT/2+I,110,157+NT/2+I,(YB(NT)+YB(NT+1))/2:NEXT
495 ONNTGOSUB5140,5130,5120,5110,5100
500 A$="":FORI=1TO32:READA:A$=A$+CHR$(A):NEXT:POSITION152,96:PATTERN-16,A$
550 IF(NP=0)+(NP>5)THEN600
560 A=.8*(160-XG(NP)):B=.8*(160-XG(NP+1)):LINE160-A,YB(NP),160-B,YB(NP+1)+2,160
+B,YB(NP+1)+2,160+A,YB(NP),160-A,YB(NP)
600 IF(NV=0)+(NV>5)THEN700
650 A=(160-XG(NV))/10:B=(YB(NV)+YB(NV+1))/2:C=(YB(NV)-YH(NV))/5:LINE160-A,B,160
,B,160-A,B-C,160+A,B-C,160,B,160+A,B
700 IF(NS=0)*(T1<>10)+(NS>5)THEN800
710 ONVGOSUB800,730,720,750:GOTO800
720 A=(160-XG(NS+1))/3:B=(YB(NS+1)-YH(NS+1))*.3:LINE160-A,YB(NS+1),160-A,YH(NS+
1)+B,160+A,YH(NS+1)+B,160+A,YB(NS+1):RETURN
730 A=(YB(NS)-YB(NS+1))/3:B=(XD(NS)-XD(NS+1))/3:A1=(YB(NS)-YH(NS))*.7-.4*A:A2=(
YB(NS+1)-YH(NS+1))*.7+3.4*A
740 LINEXD(NS)-B,YB(NS)-A,XD(NS)-B,YB(NS)-A1,XD(NS+1)+B,YB(NS)-A2,XD(NS+1)+B,YB
(NS+1)+A:RETURN
750 A=(YB(NS)-YB(NS+1))/3:B=(XG(NS+1)-XG(NS))/3:A1=(YB(NS)-YH(NS))*.7-.4*A:A2=(
YB(NS+1)-YH(NS+1))*.7+3.4*A
760 LINEXG(NS)+B,YB(NS)-A,XG(NS)+B,YB(NS)-A1,XG(NS+1)-B,YB(NS)-A2,XG(NS+1)-B,YB
(NS+1)+A:RETURN
800 CF=-T(I1-1,J1)*(V=4)-T(I1,J1+1)*(V=3)-T(I1+1,J1)*(V=2)-T(I1,J1-1)*(V=1):F=0
810 GETR:ONR+1GOTO810,810,830,810,820,850,840,810,815,810
815 GOSUB10000:PV=PV-30:GOTO810
820 V=V+1
830 V=V+1
840 V=V+1:V=V+4*(V>4):GOTO300
850 IF(T1=10)*(FC=1)*(V=3)THEN40000
860 IFCF=9THENCURSOR16,20:PRINT"  STOP!  ":PV=PV-5:GOTO810
870 IFCF=2THENPV=PV+50:GRAPHC:CURSOR5,10:PRINT" IL FAUT RENTRER MAINTENANT...."
:GOSUB4600:MM=2:FC=1:CF=0:GOSUB10000:GOSUB4600:GOTO900
880 IFCF=4THENPV=PV+50:K1=3:RESTORE11000:GOSUB4500:GOSUB10000:FORI=1TO5000:NEXT
:CF=0:GOTO900
890 IFCF=5THENPV=PV-10:T(I1,J1)=0:K1=5:RESTORE11010:GOSUB4500:IP=1-10*(RND(1)<.
5):JP=1-7*(RND(1)<.5):GOSUB200:I1=I:J1=J:T(I,J)=1:GOTO300
900 T(I1,J1)=T1:T1=CF:I1=I1+(V=4)-(V=2):J1=J1+(V=1)-(V=3):T(I1,J1)=1:PV=PV-2:FM
=FM+1:IFFM<MMTHEN300
915 T(I3,J3)=-T3*(T3<>1):I3=I3-(I1>I3)+(I1<I3):J3=J3-(J1>J3)+(J1<J3):T3=T(I3,J3
):T(I3,J3)=3:FM=0
920 IFT3=1THENGRAPHC:PRINTCHR$(6);"Le monstre du LABY vous a saute dessus":N=-1
:T1=0:GOTO1020
930 GOTO300
1000 PRINTCHR$(6);" Le monstre du LABY est a";N+1;" case";CHR$(-115*(N>0)):PRIN
TTAB(4);"et possede";FM;" points de vie":GOSUB4600:GRAPHC
1010 IF3*RND(1)<1THENK1=2:RESTORE11020:GOSUB4510:GOSUB4600:F=1:GOTO300
1020 CURSOR10,10:PRINT" IL ATTAQUE ...        ":GOSUB4600:D=INT(6*RND(1)):PRINT:PRI
NTTAB(3);"     ET FAIT UN";D+1;" AVEC LE DE          ":GOSUB4400:GOSUB4600
1030 CURSOR3,12:PRINT"Cela vous coute";(D+1)*2*(6-N);" points de vie":PV=PV-(D+
1)*2*(6-N):GOSUB4600:IFPV<0THEN30000
1040 CURSOR10,10:PRINT" A VOUS DE JOUER ":CURSOR3,12:PRINT" 1 POUR FUIR * 3 POU
R ATTAQUER  "
1050 GETR:IFR=1THENPRINT:PRINT"Tapez sur une des directions(2 4 OU 6)":PRINT"Le
5 vous rapproche du monstre..":GOTO810
1060 IFR<>3THEN1050
1070 CURSOR3,12:PRINT" APPUYEZ SUR 7 POUR LANCER LE DE "
1080 GETR:D=D+1:GOSUB4400:IFR<>7THEN1080
1090 GOSUB4400:GOSUB4600:CURSOR3,12:PRINT"Le monstre perd";(D+1)*4*(6-N);" poin
ts de vie   ":PM=PM-(D+1)*4*(6-N):GOSUB4600:IFPM>0THEN1000
1100 PRINTCHR$(6):CURSOR14,5:K1=4:RESTORE11030:GOSUB4510:T(I3,J3)=T3:IP=1:JP=8:
GOSUB200:T(I,J)=3:I3=I:J3=J:T3=0:PM=50:GOTO300
4400 D=D-INT(D/6)*6:CURSOR15,19:PRINT"┌───┐":FORI=0TO2:PRINTTAB(15);"|";MID$(D$
(D),3*I+1,3);"|":NEXT:PRINTTAB(15);"└───┘":RETURN
4500 PRINTCHR$(6):GRAPHC
4510 FORK=1TOK1:READA$:PRINTTAB(20-LEN(A$)/2);A$:PRINT:FORL=1TO1000:NEXTL,K
4600 FORL=1TO3000:NEXTL:RETURN
5000 ON8*RND(1)+1GOTO5010,5011,5012,5013,5014,5015,5016,5017
5010 RESTORE5030:RETURN
5011 RESTORE5031:RETURN
5012 RESTORE5032:RETURN
5013 RESTORE5033:RETURN
5014 RESTORE5034:RETURN
5015 RESTORE5035:RETURN
5016 RESTORE5036:RETURN
5017 RESTORE5037:RETURN
5030 DATA9,9,9,0,9,9,9,0,9,9,9,0,0,0,9,0,9,0,0,0,9,0,9,9,0,0,0,9,9,0,0,0,0,0,0,
9,0,0,9,0,9,0,9,9,0,9,0,9,9,0,9,0,0,0,0,9,0,0,0,0,9,0,9,0,9,9,9,0,9,9
5031 DATA9,9,9,0,9,9,9,0,9,9,9,0,9,0,0,0,9,0,9,0,9,0,0,0,9,0,0,0,0,0,0,0,9,9,9,
0,9,9,9,0,9,9,9,0,0,0,9,0,9,0,9,0,9,0,9,9,9,0,0,0,9,0,0,0,9,0,0,0,9,0
5032 DATA9,9,9,0,9,9,9,0,9,9,9,0,0,0,0,0,0,0,0,9,0,9,9,9,0,9,9,9,0,0,0,9,0,9,
0,9,0,9,0,9,0,9,0,9,0,9,0,9,0,9,0,9,0,9,9,9,0,9,0,9,0,0,0,0,0,0,0,0,0
```

```
 5033 DATA9,9,9,0,9,9,9,0,9,9,9,9,9,0,0,0,0,0,0,0,9,0,0,9,9,9,9,9,9,0,0,0,9,0,0,
0,0,0,0,0,9,0,9,0,9,9,9,9,9,9,9,0,9,0,0,0,0,0,9,0,9,0,0,0,9,0,9,0,0,0
 5034 DATA9,9,9,0,9,9,9,0,9,9,9,0,0,0,0,0,0,0,0,0,9,0,9,9,0,9,9,9,9,0,0,0,9,9,0,
9,0,0,0,0,9,9,0,0,0,9,0,9,0,9,9,0,0,9,0,9,0,9,0,0,9,9,9,0,0,9,0,0,9,9
 5035 DATA9,9,9,0,9,9,9,0,9,9,9,0,0,0,9,0,9,0,9,0,9,0,9,9,9,0,9,0,9,0,0,0,0,0,0,
0,0,0,9,0,9,0,0,0,9,0,9,0,0,0,9,0,9,9,9,0,9,9,0,9,9,0,9,0,0,0,9,0,0,9
 5036 DATA9,9,9,0,9,9,9,0,9,9,9,9,0,0,0,0,0,0,0,0,9,9,0,9,9,9,0,9,9,0,0,0,9,0,0,
0,9,0,0,0,9,0,9,0,9,0,9,0,9,0,9,0,9,0,9,0,9,0,9,0,9,0,0,0,9,0,0,0,9,0
 5037 DATA9 ,9,9,0,9,9,9,0,9,9,9,0,0,0,0,0,9,0,9,0,9,9,9,0,9,9,9,0,9,0,0,0,0,0,0
,0,0,0,0,0,9,0,9,0,9,0,9,0,9,9,9,0,9,0,9,0,9,0,9,0,9,0,9,0,9,0,9,0,0,0
 5100 RESTORE5200:RETURN
 5110 RESTORE5210:RETURN
 5120 RESTORE5220:RETURN
 5130 RESTORE5230:RETURN
 5140 RESTORE5240:RETURN
 5200 DATA0,0,0,0,0,0,0,0,0,0,2,2,1,0,0,0,0,0,0,0,0,0,0,128,192,160,160,192,12
8,128,128
 5210 DATA0,0,0,0,0,0,1,0,0,0,4,4,3,0,0,0,0,0,0,0,192,240,200,200,200,200,208,
224,192,192,192
 5220 DATA0,0,0,0,1,0,0,1,1,1,9,9,4,3,1,1,0,0,0,128,192,240,136,200,200,200,200,
200,144,224,192,192
 5230 DATA0,0,1,0,1,1,1,1,1,1,9,5,9,8,7,1,0,192,224,216,228,38,230,38,230,38,230
,38,228,200,240,224
 5240 DATA1,3,1,0,1,3,2,67,163,146,19,18,35,34,17,15,192,96,248,140,198,99,163,9
9,227,163,99,163,227,166,204,248
 10000 GRAPHC:PRINTCHR$(6):CURSOR9,5:FORJ=0TO15:FORI=0TO21:X=T(I,J):IFX=9THENPRI
NT"■";:GOTO10020
 10001 IFX=0THENPRINT" ";:GOTO10020
 10002 IFX=1THENPRINTCHR$(-129*(V=3)-130*(V=1)-131*(V=2)-132*(V=4));:GOTO10020
 10003 PRINTCHR$(-136*(X=2)-133*(X=3)-134*(X=4)-145*(X=5)-148*(X=10));
 10020 NEXTI:PRINT:PRINTTAB(9);:NEXTJ:RETURN
 11000 DATA"VOUS BUVEZ LA POTION MAGIQUE","QUI VOUS DONNE 50 POINTS DE VIE","ET
VOUS PERMET DE VOIR LE PLAN"
 11010 DATA"UNE TRAPPE S'OUVRE SOUS VOS PAS..","VOUS DISPARAISSEZ","DANS LE PAS
SAGE SECRET","POUR RESSORTIR EN UN AUTRE LIEU","DU LABYRINTHE"
 11020 DATA"Il semble dormir.....","Vous pouvez passer sans faire de bruit."
 11030 DATA"BRAVO!","LE MONSTRE A DISPARU...","Mais il reapparait en un autre
point","du labyrinthe"
 11040 DATA"LE MONSTRE DU LABY VOUS A VAINCU..","NE VOUS DESOLEZ PAS!","COMME VO
US D'AUTRES ONT ESSAYE EN VAIN","CE SERA POUR UNE AUTRE FOIS!!","AU REVOIR"
 11050 DATA"BRAVO!!","VOUS AVEZ MENE A BIEN VOTRE MISSION.","CELA N'ARRIVE QU'AU
X CHEVALIERS","TRES COURAGEUX","DEPECHEZ-VOUS !!"
 11060 DATA"LA PRINCESSE VOUS ATTEND AU CHATEAU."," ","BONNE SOIREE.."
 11100 DATA"VOUS ETES DANS UN LABYRINTHE","SEME D'EMBUCHES","A LA RECHERCHE DE L
'EPEE SACREE(♣)"," ","UN MONSTRE(♠) LA VEILLE","ET VA CHERCHER A VOUS DEVORER."
 11110 DATA" ","DES TRAPPES(¥) SOURNOISES","SONT PLACEES DANS LE LABYRINTHE","HE
UREUSEMENT","DES VERRES DE POTION MAGIQUE(♥)","POURRONT VOUS RECONFORTER."
 11120 DATA" ","VOTRE MISSION: RAMENER L'EPEE","A L'ENTREE DU LABYRINTHE(o).","
POUR REGARDER AUTOUR DE VOUS:","GAUCHE 4   6 DROITE","          2 DERRIERE"," "
 11130 DATA" ","POUR VOUS DEPLACER TAPER 5"," ","LA TOUCHE 8 PERMET DE VOIR LE P
LAN","MAIS VOUS COUTE 30 POINTS DE VIE.."," ","BON COURAGE!!"
 30000 RESTORE11040:K1=5:CONSOLER:PRINTCHR$(6):CURSOR0,5:GOSUB4510:GOSUB4600:GRA
PH01:CONSOLEN:PRINTCHR$(6):END
 40000 CONSOLER:LINE140,155,140,85,120,85,120,155,200,155,200,85,180,85,180,155:
LINE140,105,180,105
 40010 LINE111,189,150,155,150,130,170,130,170,155,208,189:LINE120,85,115,85,115
,75:LINE140,85,145,85,145,75:LINE180,85,175,85,175,75
 40030 LINE200,85,205,85,205,75:LINE175,75,178,75,178,78,183,78,183,75,188,75,18
8,78,192,78,192,75,197,75,197,78,202,78,202,75,205,75
 40040 LINE115,75,118,75,118,78,123,78,123,75,128,75,128,78,133,78,133,75,138,75
,138,78,143,78,143,75,145,75
 40050 FORX=140TO170STEP10:LINEX,95,X+2,95,X+2,100,X+7,100,X+7,95,X+10,95:NEXT.
 40060 A$=CHR$(28)+"*II7II7":POSITION145,115:PATTERN-8,A$:POSITION167,115:PATTER
N-8,A$:POSITION126,95:PATTERN-8,A$:POSITION186,95:PATTERN-8,A$
 40070 RESTORE11050:PRINTCHR$(6):FORI=1TO8:CURSOR0,4:PRINTSTRING$(" ",39):READA$
:PRINTCHR$(2);TAB(20-LEN(A$)/2);A$:FORJ=1TO2000:NEXTJ,I
 40080 FORI=0TO50STEP3:BLINE111-I,189,111-I,63-I,208+I,63-I,208+I,189:LINE108-I,
189,108-I,60-I,211+I,60-I,211+I,189:LINE108-I,130,120,130:LINE200,130,211+I,130
 40090 NEXT:GOSUB4600:CONSOLEN:CURSOR0,24:END
 45000 CONSOLER:PRINTCHR$(6):CURSOR0,5:GOSUB4510:GOSUB4600:GRAPHC:CONSOLEN:END
 50000 PRINTCHR$(6):GRAPHC:RESTORE11100:K1=9:GOSUB4510:GOSUB50100:PRINTCHR$(6):K
1=6:GOSUB4510:GOSUB50100:PRINTCHR$(6):K1=11:GOSUB4510:GOSUB50100:GOTO220
 50100 GETR$:IFR$=""THEN50100
 50110 RETURN
 60000 PRINTCHR$(6):CONSOLER:GRAPHC,01:LINE30,1,30,199:LINE290,1,290,199
 60010 FORY=35TO70STEP5:FORJ=30+FTO290STEP10:LINEJ,Y,J,Y+5:NEXTJ: F=-5*(F=0) :LI
NE30,Y,290,Y:NEXT:LINE30,75,290,75
 60020 FORY=75TO120STEP5:FORJ=30+FTO140STEP10:LINEJ,Y,J,Y+5:NEXTJ:LINE30,Y,140,Y
 60030 FORJ=180+FTO290STEP10:LINEJ,Y,J,Y+5:NEXTJ:LINE180,Y,290,Y:F=-5*(F=0):NEXT
:LINE30,125,290,125:LINE60,199,140,125,140,75,180,75,180,125,260,199:RETURN
```

JEUX BASSE RESOLUTION

ASTEROIDES	('PRATIQUE DU MZ 80 K')	TENTER DE TRAVERSER L'ECRAN EN EVITANT DES OBSTACLES EN TOUS GENRES. (1 JOUEUR)	[K]	*****	SP 5025	4K/oct.
ALLUMETTES	(MICRO.SYST.)	CELEBRE JEU CONTRE MZ; CELUI QUI PREND LA DERNIERE ALLUMETTE A PERDU. (1 JOUEUR CONTRE MZ)	[K][A][B]	**	SP 5025	2K/oct.
ALUNISSAGE	(?)	UNE DES MEILLEURES EXPLOITATION GRAPHIQUE DE MZ; DEMANDE BEAUCOUP DE SANG-FROID	[K]	********	SP 5025	12K/oct.
AUTO CROSS	(ANGL.)	FAIRE UN CIRCUIT EN UN MINIMUM DE TEMPS EN EVITANT TOUS LES OBSTACLES (GRAPH & SON.)	[K][A]	*****	SP 5025	5K/oct.
ANAGRAM	(A.DUBOIS)	DENOMBRE LES PERMUTATIONS POSSIBLES D'UN MOT QUELCONQUE ET LES AFFICHE.	[B]	*****	SB 5510	4K/oct.
ANNONCES BRIDGE	(A.ROUX)	INDIQUE DANS LA MAJEURE CINQUIEME;L'OUVERTURE D'ANNONCE;MAIN ALEATOIRE OU INDIQUEE..	[K][A][B]	*****	VM 5060	4K/oct.
BOWLING	(S.B.M.)	TRES GRAPHIQUE; AFFICHE LES SCORES ET LES QUILLES COMME DANS LA REALITE. (8 JOUEURS MAXI.)	[K][A]	******	SP 5025	5K/oct.
BLOCK KUSUSHI	(JAPON)	MUR DE BRIQUE TRES DIFFICILE A BATTRE; EN ASSEMBLEUR; 3 NIVEAUX	[K]	******	MACHINE	10K/oct.
BOGGLE	(A.CORNU)	VARIANTE GRAPHIQUE DU 'MOT LE PLUS LONG'; JUSQU'A 9 JOUEURS.	[K][A][B]	****	VM 5060	6K/oct.
BLACK JACK	(A.FEDIDA)	PASSIONNANT JEU DE CARTES CONTRE MZ.UTILISE TOUTES LES POSSIBILITES GRAPH & SON DU MZ	[K]	********	SP 5025	25K/oct.
COURSE DE VOITURE	(???)	ROUTE ALEATOIRE; 3 NIVEAUX DE VISIBILITE; 5 LARGEURS DE ROUTE (1 JOUEUR)	[K]	*****	VM 5060	6K/oct.
COLDITZ	(ANGLETERRE)	JEU DE SITUATION CONVERSATIONNEL EN ANGLAIS:EVASION MOUVEMENTEE DE LA FORTERESSE DE COLDITZ.	[K][A][B]	*****	SP 5025	14K/oct.
CHASSE	(ANGL.)	ATTRAPPER LE FAUVE ECHAPPE; LE PLUS RAPIDEMENT POSSIBLE	[K][A]	****	SP 5025	5K/oct.
CARRE MAGIQUE	(S.BIZOIRRE)	MZ TROUVE TOUT CARRE MAGIQUE D'UNE TAILLE MAXIMUM DE 9 * 9 NOMBRES	[B]	*****	SB 5510	8K/oct.
ECHECS MZ 50	(ANGLETERRE)	BONNES PARTIES AU NIVEAU 5 CONTRE JOUEUR MOYEN; AFFICHE LES COUPS; GRAPHIQUE REUSSI.	[K]	*******	MACHINE	48K/oct.
ECHECS SARGON 2.5	(ALLEMAGNE)	LE MEILLEUR PROGRAMME D'ECHECS SUR ORDINATEUR MAIS IL FAUDRA S'HABITUER AU GRAPH DES PIECES.	[K]	*********	MACHINE	48K/oct.
ELISA	(CLUB)	MZ RESOUDRA (EN ANGLAIS) TOUS VOS PROBLEMES PSYCHOLOGIQUES EN CONVERSANT AVEC VOUS	[K][A][B]	*****	SP 5025	6K/oct.
FREQUENCES	(ANGL.)	UN UTILITAIRE TRES PRATIQUE POUR VOUS AIDER A CREER TOUTES SORTES DE SONS SUR MZ.	[K][A][B]	******	VM 5060	3K/oct.
GALACTICAN	(ALL.)	SUPER 'SPACE INVADER'; LE MEILLEUR PROGRAMME DE JEU EN ASSEMBLEUR SUR 80K; SAISISSANT..	[K]	*********	MACHINE	8K/oct.
HANOI MODIFIE	(ANGL.)	LE JEU COMMENCE AVEC UNE COMBINAISON PRE-ETABLIE; 5 NIVEAUX.	[K][A]	*****	VM 5060	4K/oct.
HEAD ON	(ANGL.)	UN CLASSIQUE; DEMANDE BEAUCOUP DE REFLEXES..ET DE CALME CAR GAGNER EST UN EXPLOIT !.	[K]	*******	MACHINE	6K/oct.
ISOLA	(ANGLETERRE)	SE JOUE A DEUX; LE PREMIER QUI ARRIVE A IMMOBILISER SON ADVERSAIRE A GAGNE.	[K][A][B]	*****	VM 5060	4K/oct.
JUMPING BALLL	(S.B.M.)	JEU DE REFLEXION TRES RAPIDE	[K][A][B]	***	SP 5025	2K/oct.
JACQUET	(?)	JEU CONTRE MZ AVEC GRAPHISME DU JEU ET DES DES.	[K]	*******	VM 5060	15K/oct.
JUNTORY GAMES	(ANGL.)	IDEM A 'ISOLA' MAIS EN ASSEMBLEUR DONC BEAUCOUP PLUS RAPIDE ET MIEUX SONORISE.	[K]	*******	MACHINE	4K/oct.
JEU DE LA VIE (B)	(J.BERCOVICI)	TRES GRANDE RAPIDITE D'EXECUTION DU A UN SOUS PROGRAMME EN ASSEMBLEUR.	[B]	******	SB 5510	40K/oct.
JEU DE LA VIE (K/A)	(CLUB)	TRES RAPIDE	[K][A]	******	MACHINE	3K/oct.
KINGDOM	(?)	SI VOUS DESIREZ REGNER LONGTEMPS SUR VOTRE ILE	[K][A][B]	*****	VM 5060	11K/oct.
LABYRINTHE EN 3 DIMENSIONS	(S.B.M.)	SORTIR D'1 LABYRINTHE DONT ON CHOISIT LES DIMENSIONS; GRAPH.& PASSIONNANT.	[K]	*******	VM 5060	20K/oct.
LE MARAIS DE L'ALLIGATOR	(ANGLETERRE)	CUEILLIR LES NOIX DE COCO SANS SE FAIRE MANGER PAR L'ALLIGATOR;GRAPH & SONS TRES REUSSIS	[K]	*******	SP 5025	9K/oct.
LE VOLEUR	(ANGL.)	VOUS ETES LE VOLEUR ET DEVEZ RAMASSER LE MAXIMUM DE DIAMANTS EN EVITANT LES CHIENS.	[K][A]	*****	SP 5025	5K/oct.
LE CHATEAU HANTE	(?)	JEU DE SITUATION TRES COMPLET; PEUT DURER PLUSIEURS HEURES !; LIVRE AVEC NOTICE..	[K][A]	********	VM 5060	25K/oct.
LA BOITE A ATOMES	(S BIZOIRRE)	JEU DE REFLEXION; TRES DIFFICILE A BATTRE AU NIVEAU 9; GRAPHIQUE.	[B]	******	SB 5510	64K/oct.
LAS VEGAS	(?)	FIDELE RECONSTITUTION D'UNE MACHINE A SOUS;TRES BEAUX GRAPHS.:1/2 BASIC-1/2 ASSEMBLEUR	[K]	********	SP 5025	10K/oct.
MUR DE BRIQUE	(ANGLETERRE)	JEU D'ADRESSE; GRAPHIQUE; NIVEAU MOYEN.	[K]	****	VM 5060	6K/oct.
MORPION	(L'O.I.)	ALLIGNER 5 CROIX DANS UNE GRILLE 10 * 10; MZ LENT (30 sec.) MAIS TRES FORT.	[K][A][B]	*****	VM 5060	5K/oct.
MISSILES	(?)	TENTER D'ATTEINDRE 10 CIBLES SUR GRILLE DE 10*10 AVEC INDICATIONS A CHAQUE COUP TIRE.	[K][A]	****	VM 5060	5K/oct.
MINEUR	(?)	VOUS CHERCHEZ DE L'OR DANS UNE MINE ET DEVEZ AFFRONTER MOULTE OBSTACLES; BEAU GRAPH.	[K]	******	VM 5060	9K/oct.
MAISON MYSTERIEUSE	(COQUEREZ)	MEFIEZ VOUS DE CETTE MAISON OU VOUS ATTENDENT 1000 PERIPETIES OU...LA MORT!	[K][A][B]	******	SP 5025	15K/oct.
OTHELLO 1	(S.B.M.)	NIVEAU MOYEN.GRAPHIQUE.	[K][A][B]	****	SP 5025	5K/oct.
OTHELLO 2	(A.URCUN)	MOINS GRAPHIQUE MAIS BEAUCOUP PLUS FORT QU' OTHELLO 1.	[K][A][B]	******	SP 5025	7K/oct.
OBSTACLES	(R.GERMAIN)	TRAVERSER L'ECRAN EN EVITANT UNE MULTITUDE D'OBSTACLES. PROG. EN BASIC ET ASSEMBLEUR.	[B]	****	SB 5510	6K/oct.
POKER	(S.B.M.)	BEAU GRAPHISME DES CARTES. NIVEAU MOYEN.	[K]	*****	SP 5025	6K/oct.
PRISONNIERS	(A.CORNU)	MZ TENTE DE SORTIR D'UN LABYRINTHE DONT VOUS AUREZ CHOISI LA DIFFICULTE; TRES DEMONSTRATIF.	[K][A]	******	VM 5060	3K/oct.
PAPER STONE	(?)	JEU DU PAPIER-PIERRE-CISEAUX...EROTIQUE..	[K][A]	*****	VM 5060	8K/oct.
POULES ET RENARD	(ANGLETERRE)	IL FAUDRA BEAUCOUP DE RUSE AU RENARD (VOUS) POUR ECHAPPER AU RENARD (MZ) SUR UN DAMIER.	[K][A]	******	SP 5025	5K/oct.
PUISSANCE 4	(?)	JEU DE DEDUCTION CONTRE MZ; DIFFICILE A BATTRE (AU DEBUT...); JOLI GRAPHISME.	[K][A]	*****	VM 5060	8K/oct.



RAIL	(?)	PAS FACILE ET PASSIONNANT DE FAIRE TRAVERSER LA LOCOMOTIVE EN COMMANDANT LES AIGUILLAGES.	[K][A][B]	*****	VM 5060	7K/oct.
RHINOCEROS	(J.ANCHISE)	IL FAUT VOUS SAUVER DE LA FORET AVANT QUE LE RHINOCEROS NE VOUS ATTRAPPE (GRAPH)	[K][A]	*****	VM 5060	6K/oct.
RETOURNE	(P.BELMONDO)	VERSION SPECIALE MZ D'UN JEU TRES CONNU; DEMANDE BEAUCOUP DE MEMOIRE VISUELLE.GRAPHIQUE	[K][A][B]	*****	SP 5025	6K/oct.
SPACE FIGHTER	(S.B.M.)	JEU DE REFLEXION ET D'ADRESSE. RAPIDE; GRAPHIQUE; SONORE	[K]	******	SP 5025	6K/oct.
STAMP OUT	(ANGLETERRE)	JEU D'ADRESSE; ATTRAPER LE MAXIMUM D'ETOILES EN EVITANT DE + EN + D'OBSTACLES. (1 JOUEUR)	[K][A]	*****	VM 5060	5K/oct.
STARTRECK	(A.FEDIDA)	CELEBRE JEU DE GUERRE SPATIALE; UTILISE TOUTES LES POSSIBILITES GRAPHIQUES & SONORES DU MZ.	[K][A][B]	********	VM 5060	20K/oct.
SOUS MARIN	(REVUE)	VOTRE SONAR VOUS AIDERA A COULER DES NAVIRES SUR UNE GRILLE DE DIMENSIONS VARIABLES.	[K]	*****	VM 5060	6K/oct.
SUPER MASTERMIND	(REVUE)	COMBINAISONS DE 3;4;5 OU 6 CHIFFRES. 1 JOUEUR CONTRE MZ OU 2 JOUEURS.	[K][A][B]	*****	VM 5060	6K/oct.
SUPER LOGOTRON	(REVUE)	CREATION MZ'ESQUE D'UN NOMBRE INFINI DE MOTS AVEC LA SIGNIFICATION DE LEUR RACINE.	[K][A][B]	*****	VM 5060	48K/oct.
SAUVETAGE	(REVUE)	VOUS DEVEZ ATTRAPER LE MAXIMUM DE PERSONNES QUI SAUTENT D'1 IMMEUBLE EN FEU.(GRAPH & SON)	[K]	*******	SP 5025	16K/oct.
STRIP-TEASE	(J.P.GLUCK)	AU DEBUT DU JEU; LA JEUNE FILLE EST HABILLEE; LA SUITE VOUS COUTERA 6 POINTS..	[B]	******	SB 5510	9K/oct.
TOURS DE HANOI	(???)	JEU DE (GRANDE) PATIENCE CHINOIS; DEMANDE BEAUCOUP DE REFLEXION. (1 JOUEUR)	[K][A][B]	*****	SP 5025	4K/oct.
TELECRAN	(S.BIZOIRRE)	CREATION DE DESSINS SUR L'ECRAN AVEC LA FONCTION 'SET' (JEU POUR ENFANT).	[K][A][B]	***	SP 5025	3K/oct.
TELECRAN MEMOIRE	(A.FEDIDA)	IDEM A TELECRAN MAIS AVEC CARACTERES NORMAUX ET MEMORISATION DE PLUSIEURS PAGES DE DESSIN.	[K]	*****	SP 5025	6K/oct.
TOUR INFERNALE	(JAPON)	UNE TOUR EN FLAMMES..ET VOUS DEDANS; PAS FACILE DE S'EN SORTIR. BEAU PROG.;GRAPH & SON.	[K]	********	MACHINE	10K/oct.
TIERCE	(?)	VOUS MISEZ;MZ VOUS DONNE LES COTES;VOUS FAIT ASSISTER A LA COURSE;ET DISTRIBUE LES GAINS.	[K]	*******	VM 5060	12K/oct.

JEUX HAUTE RESOLUTION

DEMO H.R.	(CLUB)	COURTE MAIS SAISISSANTE DEMO DE HAUTE RESOLUTION (GRAPH1)	[B]	*****	SB 5510	3K/oct.
FIGHTER	(S.B.M.)	ABATTRE LE MAXIMUM D'AVIONS QUI TRAVERSENT L'ECRAN. (CARTES GRAPH. 1 ET 2).	[B]	****	SB 5510	6K/oct.
FANCY CLOCK	(S.B.M.)	LE SUMMUM DE L'HORLOGE MURALE NUMERIQUE ET A AIGUILLES.(GRAPH 1).	[B]	*******	SB 5510	12K/oct.
GRAPHISMES	(J.P.GLUCK)	8 GRAPHISMES DIFFERENTS:COCON	[B]	******	SB 5510	4K/oct.
LAS VAGAS-B	(S.B.M.)	IDEM 'LAS VEGAS-K'AVEC GRAPH H.R. SAISISSANTS ! (GRAPH 1).	[B]	********	SB 5510	40K/oct.
LABYRINTHE 3 DIM	(J.MILLET)	TRES BELLE VERSION POUR MZ 80 B.PASSIONNANTE;BEAUX GRAPHS (GRPH1)	[B]	********	SB 5510	64K/oct.
OTHELLO H.R	(J.P.LHOIR)	UN PROG. ORIGINAL ET TRES FORT.	[B]	********	SB 5510	40K/oct.
POUSSE-POUSSE	(S.BIZOIRRE)	REPLIQUE DU CELEBRE JEU DE PATIENCE; 5 NIVEAUX;DEMANDE PATIENCE & LOGIQUE; GRAPHIQUE ANIME.	[B]	********	SB 5510	40K/oct.
PENDU H.R	(CLUB)	VOUS AUREZ UN BEAU PENDU EN H.R MAIS DEVREZ CREER VOTRE PROPRE LISTE DE MOTS.(GRAPH1)	[B]	******	SB 5510	10K/oct.
PETALES DE VIE	(L'O.I.)	ASTROLOGUE & ASTRONOME; MZ VOUS PREDIRA l'AVENIR SUR 1 AN AVEC UN TRES BEAU GRAPH (GR1)	[B]	*******	SB 5510	20K/oct.
ROSACE	(J.P.GLUCK)	CREATION D'1 VITRAIL POUR St. SHARP;CHAPELLE DE'AIMEZED';LIEU DIT 'QUATREVINGTBE'(GPH 1)	[B]	******	SB 5510	12K/oct.
SPACE ODYSSEY	(ANGL.)	JEU GRAPHIQUE H.R. PASSIONNANT SI L'ON EST HABILE ET PATIENT. BEAUX GRAPHS. (GRAPH 1).	[B]	********	SB 5510	64K/oct.

UTILITAIRES

BASICOIS	(L'O.I.)	BASIC EN FRANCAIS POUR BASIC 5010 ET 5025	[K]	***	SP 5025	2K/oct.
COPIE K7	(ANGL.)	VOUS PERMET DE RECOPIER TOUT PROGRAMME ECRIT EN ASSEMBLEUR.	[K]	******	MACHINE	1K/oct.
C.A.O. DESSIN GRAPH	(A.CORNU)	AIDE A LA CREATION DE DESSINS ET INSERTION AUTOMATIQUE DANS UN PROGRAMME	[K]	****	SP 5025	20K/oct.
DESASSEMBLEUR	(REVUE)	PERMET DE TRADUIRE; EN CLAIR; TOUTES LES INSTRUCTIONS Z 80 CONTENUES DANS UN PROG. MACHINE	[K][A][B]	*******	VM 5060	18K/oct.
DESASSEMBLEUR MACHINE	(KOKANOSKI)	DESASSEMBLEUR EN LANG.MACHINE;ECRAN OU IMPRIMANTE;+5 FOIS + RAPIDE MAIS NON MODIFIABLE	[K][A]	*******	MACHINE	4K/oct.
KOKAN.SUPER BASIC 2	(B.KOKANOWSKI)	VOIR No.1 POUR EXPLICATIONS ET CE No. POUR COMPLEMENT D'INFORMATIONS.	[K]	**********	MACHINE	18K/oct.
MODIFS 5025	(B.KOKANOSKI)	MODIFIE ET AMELIORE LES POSSIBILITES DU SP 5025. VOIR ARTICLE BULLETIN No.2	[K]	****	MACHINE	1K/oct.
PATTERN MAKER	(CLUB)	UN UTILITAIRE QUI FACILITERA LA CREATION DE VOS GRAPHISMES H.R.(GRAPH1).	[B]	******	SB 5510	5K/oct.
RENUM	(J.P.GLUCK)	PERMET LA RENUMEROTATION DES LIGNES D'UN PROG.; SANS MODIFIER LES GOTO; GOSUB; ETC..	[B]	****	SB 5510	3K/oct.

MUSIQUE, SONS ET IMPRIMANTE

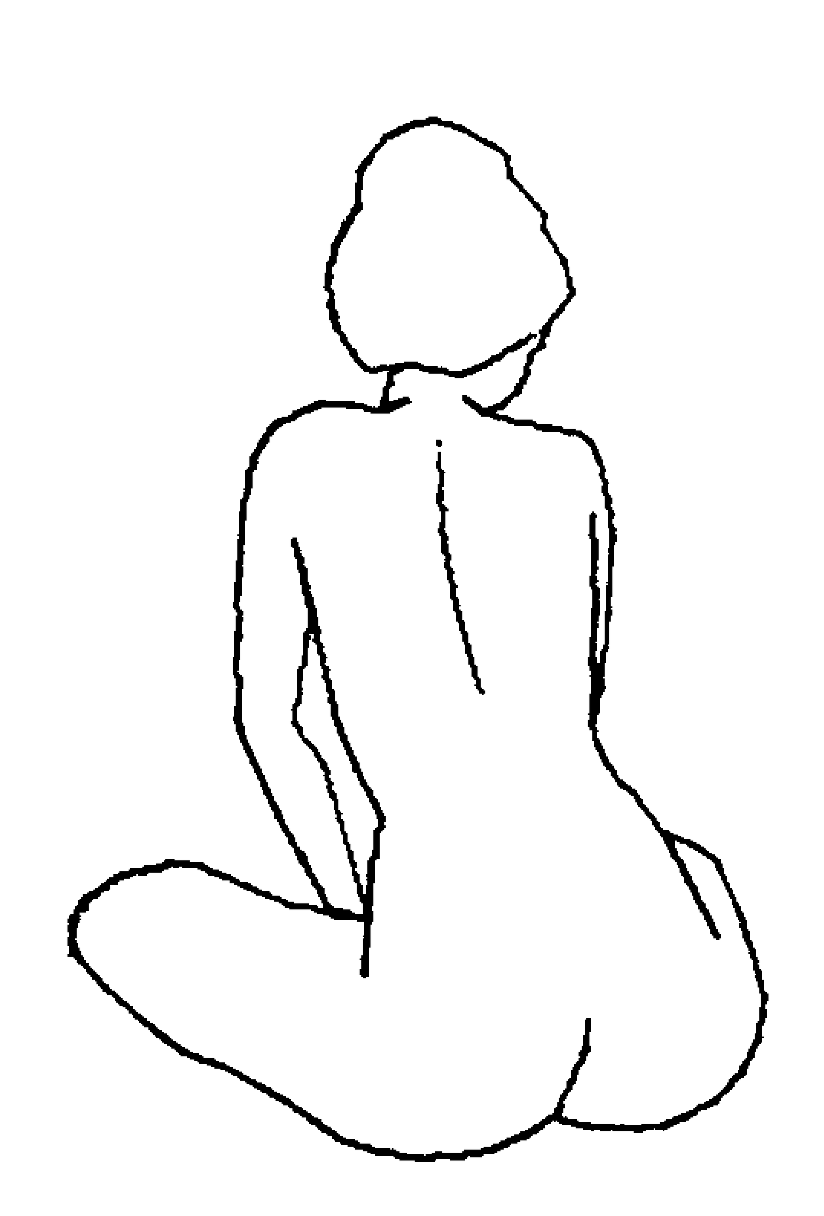
ACCORDEON	(S.BIZOIRRE)	'LE VOL DES HIRONDELLES'; UN P'TIT AIR D'MUSETTE.	[K][A][B]	***	SP 5025	4K/oct.
BACH	(S.BIZOIRRE)	2 EXTRAITS DES SUITES POUR VIOLONCELLE (SYNTHETIQUE)	[K][A][B]	****	SP 5025	8K/oct.
COPY P5 * 2	(S.BIZOIRRE)	PERMET UNE RECOPIE D'ECRAN BASSE OU HAUTE RESOLUTION SUR IMPRIMANTE EN GRAND FORMAT.	[B]	******	SB 5510	3K/oct.
MUSIC-B	(A.DUBOIS)	PERMET DE JOUER LES 3 GAMMES DIRECTEMENT AU CLAVIER; LES NOTES APPARAISSENT SUR UNE PORTEE	[B]	*****	SB 5510	7K/oct.
MUSIC 4	(CLUB)	MZ CREE UNE MUSIQUE ALEATOIRE MAIS REPONDANT AUX REGLES DE LA COMPOSITION.	[K][A][B]	****	VM 5060	5K/oct.
MONA LISA	(S.BIZOIRRE)	FIDELE REPRODUCTION DE 'LA JOCONDE' SUR IMPRIMANTE P5	[B]	*******	SB 5510	15K/oct.
MUSIQUE SUISSE	(CLUB)	4 MINUTES DE TRES BELLE MUSIQUE.	[K][A][B]	*****	SP 5025	8K/oct.

ENSEIGNEMENT

CHAINE L.S.E.	SIMULATION DE TOUTES LES FONCTIONS CHAINE DU LANGAGE L.S.E. (12 SOUS-PROGRAMMES).	(A.DUBOIS)	[B]	******	SB 5510	0K/oct.
DERIVEES	DONNE LES DERIVEES SUCCESSIVES D'UN POLYNOME.	(A.DUBOIS)	[K][A][B]	****	SB 5510	3K/oct.
EQUATIONS	RESOLUTION D'EQUATIONS PAR METHODE DU PIVOT	(M.FAHRI)	[K][A][B]	*****	VM 5060	4K/oct.
GUITARE	COURS DE GUITARE COMPLET AVEC TABLATURE DES ACCORDS	(?)	[K]	********	VM 5060	48K/oct.
LECTURE C.M	ENTRAINEMENT A LA LECTURE POUR ENFANTS DE COURS MOYEN.	(F.DENIZET)	[K][A][B]	******	VM 5060	48K/oct.
LECTURE C.E	AIDE A LA LECTURE POUR ENFANTS DE COURS ELEMENTAIRE ET MOYEN.	(F.DENIZET)	[K][A][B]	******	VM 5060	48K/oct.
MULTIPREC	TRAITE EN MULTIPRECISION ADDITION	(A.DUBOIS)	[K][A][B]	****	SB 5510	3K/oct.
MOYENNES	GESTION DES NOTES D'UNE CLASSE AVEC TABLEAUX ET HISTOGRAMMES	(J.MILLET)	[B]	*******	SB 5510	64K/oct.
PASCAL	ETUDE DU TRIANGLE DE PASCAL JUSQU'A L'ORDRE 46 (SATURATION)	(A.DUBOIS)	[K][A][B]	****	SB 5510	5K/oct.
POLYGONES	MZ CREE; A LA DEMANDE; TOUTES SORTES DE POLYGONES EN H.R.(GRAPH1).	(S.BIZOIRRE)	[B]	****	SB 5510	3K/oct.
RESOLUTION EQUATIONS	EQUATIONS 1er.; 2eme.; 3eme. DEGRE + 3 EQUATIONS A 3 INCONNUES.	(A.CORNU)	[K][A][B]	****	SP 5025	3K/oct.
RACINES POLYNOME	RECHERCHE DES RACINES D'UN POLYNOME...(BIEN SUR)	(?)	[K][A][B]	****	VM 5060	3K/oct.
STATISTIQUES	STATISTIQUE DESCRIPTIVE (POUR CLASSES DE LYCEE)	(A.DUBOIS)	[K][A][B]	****	SB 5510	4K/oct.

DIVERS

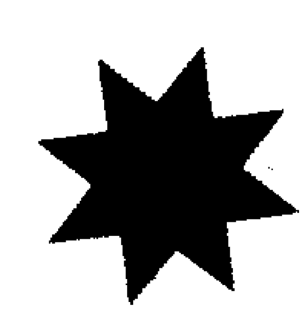
ANNUAIRE 'MICRO SYSTEMES'	TOUS LES ARTICLES DE 'MICRO SYSTEMES' REFERENCES PAR NUMERO ET PAR THEMES.	(S.BIZOIRRE)	[K][A][B]	*******	VM 5060	48K/oct.
ANNUAIRE 'ORDINATEUR INDIVIDUEL'	IDEM A ANNUAIRE 'MICRO SYSTEMES'	(S.BIZOIRRE)	[K][A][B]	*******	VM 5060	48K/oct.
ESCALIER	M.Z. ETUDIE ET CALCULE POUR VOUS LE MEILLEUR ESCALIER D'APRES UNE PENTE DONNEE.	(P.JARDE)	[K][A][B]	*****	VM 5060	5K/oct.
PARISCOP	POUR FANA DE CINE.FICHIER FILMS & SALLES;RECH.PAR CATEGORIES;REALISAT.;AN;PAYS;ACTEURS.	(CLUB)	[K][A][B]	********	VM 5060	48K/oct.
R.L.C. SERIE ET PARALLELE	DETERMINE LES VALEURS D'UN CONDENSATEUR; D'UNE RESISTANCE; D'UNE SELF DANS UN CIRCUIT.	(?)	[K][A][B]	***	SF 5025	3K/oct.



Dans le prochain bulletin,
je me retourne
(Dossier réalisé sur PC 1500)

Réalisation TD Publicité.

GAGNEZ 2603,27 F*
grâce à votre PC 1211



*** Sur l'achat d'un MZ 80A**

Du 1er janvier 1983 au 31 mars 1983, S.B.M. vous propose le MZ 80A au PRIX EXCEPTIONNEL CLUB PC 1211 de 7116,00 T.T.C. au lieu de 9719,27 F en échange de votre PC 1211 et de ses périphériques.



Caractéristiques techniques MZ-80A

Unité centrale	Z-80
Mémoire	MEM de 4K octets; MEV standard de 32K octets (extensible à 48K en option)
Affichage	Ecran cathodique vert de 9 pouces (23cm); matrice 8 × 8 points; 40 caractères sur 25 lignes
Cassette	Commandes manuelles; Cassette audio standard Transfert de données (Système PWM Sharp); 1200 bits/second
Sortie son	400 MW max. (440Hz)
Touches	Clavier ASCII; Majuscules; Minuscules symboles graphiques; pavé numérique
Gestion d'écran	Commandes du curseur (haut, bas, gauche, droite, origine, effacement); Touches d'insertion et de suppression.

COMMENT PARTICIPER :

- Vous êtes membre du club et vous posséder une PC 1211 et ses périphériques en parfait état de marche *(achetée avant le 1.12.82).*
- Renvoyez-nous le bon ci-dessous dûment rempli accompagné de votre règlement et des justificatifs d'achat de votre PC 1211 et de ses périphériques *(double de facture).*
- Vous recevrez par retour un bon d'enlèvement vous permettant de retirer votre MZA chez le distributeur régional S.B.M. le plus proche de votre domicile.
- En allant chercher votre MZ remettez le PC 1211 et ses périphériques au distributeur S.B.M. *(celui-ci se réserve le droit d'annuler l'opération au cas ou la PC 1211 n'est pas en état satisfaisant).*

ET AUSSI DES PRIX SUPER CLUB sur les périphériques

MZ 80 A	T.T.C.	au lieu de
Panier interface (AEU)	1700,00	2.352,00
Imprimante MZ 80 P5	6700,00	9055,20
Interface Floppie (AFI)	1700,00	2352,00
Boîtier disquette 1 Drive double face, double densité (MZ 80 FD1)	6500,00	8582,44

BON DE COMMANDE *à DECOUPER et à RETOURNER à* S.B.M. CLUB PC 1211
151-153, avenue Jean-Jaurès - 93307 AUBERVILLIERS — avant le 31.03.1983

Je soussigné(e).. possesseur de la PC 1211 n°..............................

AGE : .... Adresse : ....

....

Ville : .... Code postal : ....

désire bénéficier des conditions exceptionnels club pour l'achat d'un MZ 80A au prix de .... **7116F T.T.C.**

J'en profite également pour vous commander :

☐ Panier interface.. au prix de **1700,00 F T.T.C.**
☐ Imprimante P5... au prix de **6700,00 F T.T.C.**
☐ Interface Floppie .... au prix de **1700,00 F T.T.C.**
☐ Boîtier disquette - 1 Drive double face, double densité .... au prix de **6500,00 F T.T.C.**

Soit au total de francs ....

dont je joins le règlement par : ☐ CHEQUE BANCAIRE ☐ C.C.P. ☐ MANDAT

Matériel rendu : ☐ **PC 1211** ☐ **CE 121** ☐ **CE 122**

J'ai bien noté que je remettrais en échange ma PC 1211 et ses périphériques au distributeur régional S.B.M. chez qui je prendrais possession de mon MZ 80 A.

Nom et adresse *(éventuel)* du distributeur chez lequel j'aimerais être livré. ....

.... SIGNATURE :

DATE : ....